Project no.:	**027657**
Project full title:	**Perception, Action & Cognition through learning of Object-Action Complexes**
Project Acronym:	**PACO-PLUS**
Deliverable no.:	**D4.1.1**
Title of the deliverable:	**Multi-sensory Gestalts**

Contractual Date of Delivery to the CEC:	31 January 2007
Actual Date of Delivery to the CEC:	30 January 2007
Organisation name of lead contractor for this deliverable:	AAU

Author(s): Norbert Krüger, Florentin Wörgötter, Tamim Asfour, Rüdiger Dillmann, Justus Piater, Danica Kragic, Aleš Ude, Alex Bierbaum, Dirk Kraft, Morten Kjaergaard, Sinan Kalkan, Nicolas Pugeault, and Renaud Detry
Participant(s): AAU,BCCN,JSI

Work package contributing to the deliverable:	WP1, WP2, WP5.2
Nature:	R
Version:	Draft
Total number of pages:	20
Start date of project:	1st Feb. 2006 **Duration:** 48 month

Project co-funded by the European Commission within the Sixth Framework Programme (2002–2006)

Dissemination Level

PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract:

In this deliverable, we describe the work done in the context of WP 4.1. First, we introduce a number of relations defined on purely visual primitives that are used to derive visual Gestalts. These relations are used also to define first OACs but also to compute visual entities that can have an haptic equivalent. We then introduce the concept of a multi-sensorial primitive that combines visual and touch information and we show that shape and other object properties such as weight and elasticity can be derived by haptic information. We then describe a pre-grasping strategy that does not require any object models but makes use of the visual relations defined before. Finally, we describe a module that extracts multi-modal representations of objects by interaction of the robot and the visual system. The sub-modules introduced here are used in particular in the context of demo 1 (see Deliverable D8.1.1 and D8.1.2) where a robot–vision system is described which explores the environment.

Keyword list: Haptic Sensor, Gestalt Laws, Grasping, Exploration

Table of Contents

1. Introduction

Note: In this deliverable, we rather briefly describe different pieces of work that are relevant for the tasks in WP 4.1 with the intention to present a 'red thread' and the relations between them. These works are described in more detail in accepted [A, C, E, H, I, K] or reports [D, B, G, F, J] that will be the basis for future submissions.

In the spirit of Gestalt Psychology (see, e.g., [12, 11]), we define a set of (local) entities across which strong relations/predictions can be defined as a Gestalt. Commonly used relations in vision are co-linearity (or 'Good Continuation'), similarity, proximity or symmetry etc. However, Gestalt laws can also be defined for other senses and across senses.

In our work such relations are defined within one sensorial modality (the visual relations we make use of are described in Section 2.2 and [B, G]) or integrate information across different sensorial modalities (see, e.g., [A, H]). In our context, other sensorial information than vision is the proprioceptive information of a robot arm and haptic information in terms of (1) proprioceptive information of the robot gripper and (2) tactile information gathered by two sensors (described in Section 4 and [F]).

Multi-sensorial Gestalts are closely linked to the concept of object-action complexes (OACs) as becomes clear in their application in Deliverable D8.1.1 and D8.1.2. For example the 'grasping reflex' described in Section 5 is based on constellations of local visual 3D features with certain relations between them that trigger an early OAC that test a grasping hypothesis by haptic feedback. Furthermore, they are building blocks of more complex OACs (see Section 6 and Deliverable D8.1.1).

In this deliverable, we introduce 4 different kinds of Gestalts that are used in the context of the PACOplus project. In Section 3, we introduce two purely visual Gestalt that are based on visual relations of multi-modal primitives [G] [14] such as co-planarity, co-linearity and co-colority (see [B]). In Section 4, we extend the concept of a visual multi-modal primitive to a multi-sensorial primitive by adding haptic information gained by tactile sensor technology comercially used as MicroJoysticks in Laptops and we introduce a haptic Primitive (Task 4.1.2). We show that this sensor has the potential to give information about important object aspects such as surface nomals, elasticity of surfaces as well as the weight of objects. Based on the visual primitives and the relations defined upon them, we can form visual entities (called monos, see Section 3.2) that can have a haptical counterpart. This finally leads to the concept of a multi-sensorial primitive (see Section 4). The multi-sensorial primitives become extracted by and can trigger themselves explorative behavioural components (BCs) that are described in Section 4.4. Note that these components, although in the beginning not OACs, lead in a natural way to OACs when predictions associated to the BCs become reflected upon.

Central to demo 1 (see Deliverable 8.1.1 and 8.1.2) are two multi-sensorial Gestalts (described in Section 5 and 6) that combine the rich visual representation introduced in [G] [14] with the concept of grasping and rigid motion. In Section 5, we introduce a initial grasping behaviour (task 4.1.1) with which the robot can achieve physical control over unknown objects (see [A]) by relating features constellations that are co-linear and co-planar to grasping actions. In the context of task 4.1.3, an active aquisition of object representations is described in Section 6. In this process, the object itself emerges and segments itself from the background by the predictions that are based on proprioceptive information (more specifically, the self-induced motion of the grasper). Hence, the object becomes established as the set of visual features that transform according to the motion of the robot (i.e., that are related by a deterministic transformation 'rigid body motion' (RBM)).

2. An early cognitive Vision System: Primitives and their Relations

2.1 Introducing visual primitives

The primitives we will be using in this work are local, multi-modal edge descriptors that were introduced in [13] (see Figure 1). A detailed technical description of these representations is given in [G]. These primitives focus on giving a semantically and geometrically meaningful description of the local image patch. The importance of such a semantic grounding of features for a general purpose vision front-end, and the relevance of edge-like structures for this purposes were discussed in [6].

The primitives are extracted sparsely at locations in the image that are the most likely to contain edges. The sparseness is assured using a classical winner take all operation, insuring that the generative patches of the primitives do not overlap. Each of the primitive encodes the image information contained by a local image patch. Multi-modal information is gathered from this image patch, including the position $\boldsymbol{m}$ of the centre of the patch, the orientation θ of the edge, the phase ω of the signal at this point, the colour $\boldsymbol{c}$ sampled over the image patch on both sides of the edge, the local optical flow $\boldsymbol{f}$ and the size of the patch ρ. Consequently a local image patch is described by the following multi-modal vector:

$$\boldsymbol{\pi} = (\boldsymbol{m}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T, \tag{1}$$

that we will name *2D primitive* in the following. The primitive extraction process is illustrated in Fig. 1.

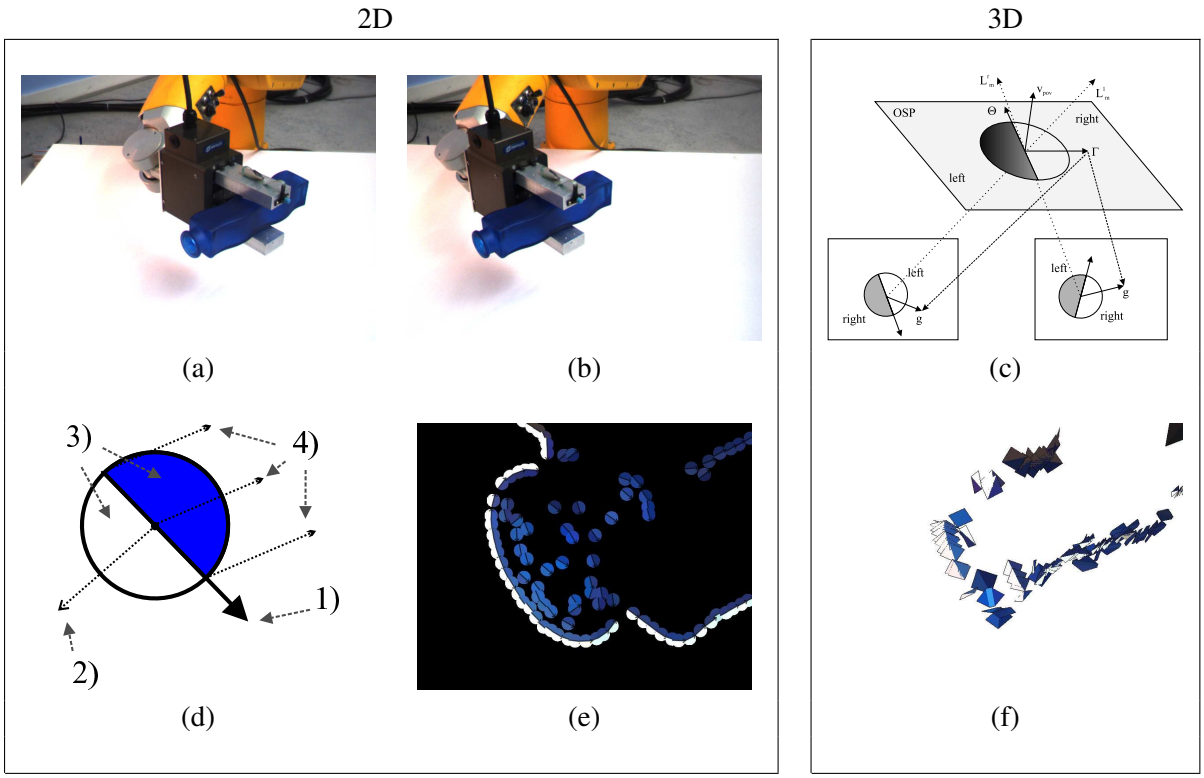
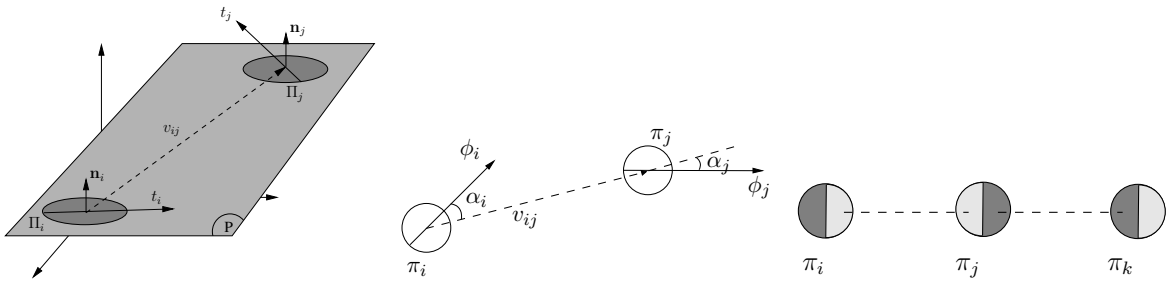


Figure 1: Overview of the system. (a)-(b) images of the scene as viewed by the left and right camera at the first frame. (d) symbolic representation of a primitive: wherein 1) shows the orientation, 2) the phase, 3) the colour and 4) the optic flow of the primitive. (e) 2D-primitives of a detail of the object. (c) reconstruction of a 3D-primitive from a stereo-pair of 2D-primitives. (f) 3D-primitives reconstructed from the scene.

In a stereo scenario, a *3D primitive* can be computed from correspondences of 2D primitives (see Fig.1)

$$\boldsymbol{\Pi} = (\boldsymbol{M}, \Theta, \boldsymbol{\Omega}, \boldsymbol{C})^T, \tag{2}$$



(a) (b) (c)

Figure 2: **(a)** Co-planarity of two 3D primitives. **(b)** Co-linearity of two 2D primitives. **(c)** Co-colority of three 2D primitives π_i, π_j and π_k. In this example, π_i and π_j are cocolor, so are π_i and π_k; however, π_j and π_k are not cocolor.

where $\boldsymbol{M}$ is the position in space, Θ is the 3D orientation, $\boldsymbol{\Omega}$ is the phase of the contour and $\boldsymbol{C}$ is the colour on both sides of the contour. We have a projection relation

$$\mathcal{P} : \boldsymbol{\Pi} \rightarrow \boldsymbol{\pi} \tag{3}$$

linking 3D-primitives and 2D-primitives.

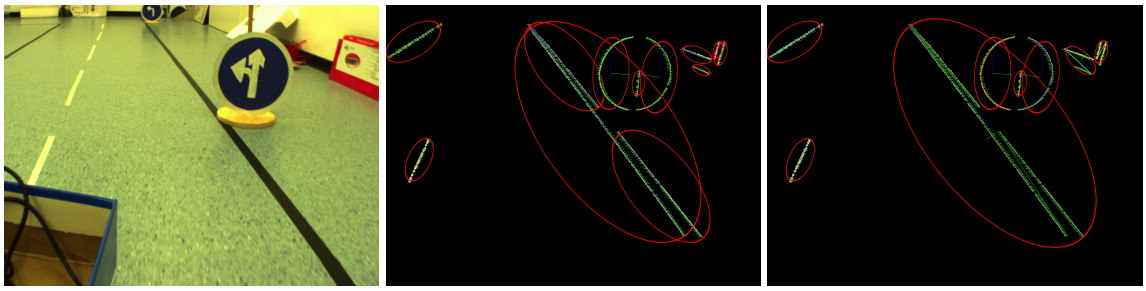
2.2 Relations on Primitives

As said before, Gestalts are defined as sets of related primitives. The sparse and symbolic nature of primitives allows the following relations to be defined on them. For more information about relations between primitives, see [B].

- *Co-planarity:* Co-planarity is defined only between 3D primitives. Two 3D edge primitives $\boldsymbol{\Pi}_i$ and $\boldsymbol{\Pi}_j$ are co-planar iff their orientation vectors t_i and t_j lie on the same plane. The co-planarity relation is illustrated in Fig. 2(a).
- *Co-linearity:* Two 2D primitives π_i and π_j are co-linear iff they are part of the same contour. Due to uncertainty in the 3D reconstruction process, in this work, the co-linearity of two spatial primitives $\boldsymbol{\Pi}_i$ and $\boldsymbol{\Pi}_j$ is computed using their 2D projections π_i and π_j.
- *Co-colority:* Two 3D primitives π_i and π_j are co-color iff their parts that face each other have the same color. In the same way as co-linearity, co-colority of two spatial primitives $\boldsymbol{\Pi}_i$ and $\boldsymbol{\Pi}_j$ is computed using their 2D projections π_i and π_j. In Fig. 2(c), a pair of co-color and not co-color primitives are shown.
- *Rigid Body Motion:* Assuming a calibrated camera system, correct correspondences between primitives in the left and right motion and knowledge about the motion of objects and the camera from one frame to the other we can predict the change of appearance of a 3D primitive (and after projection also of the corresponding 2D primitives) from one frame to the other by an explicit formula (for details, see [H]).

3. Example 1: Visual Gestalts

Note: This Section described work that has been mainly performed in another project (called Drivsco [10]). Since there are strong complementary aspects between PACOplus and Drivsco, we briefly present the results here.



(a) (b) (c)

Figure 3: Example of the extraction of Visual Gestalts using good continuation (b) and parallelism (c) (the red ellipses show the Gestalts).

3.1 Visual Gestalts

We define sets of multi-modal primitives that are linked by the relations co-linearity, co-planarity, co-colority and rigid motion as *Visual Gestalts*. If the motion estimation comes from another sensor — for example from the proprioceptive information of the robot movements (as in Section 6) — they are called a *multi-sensory Gestalt*.

The simplest kind of Visual Gestalt we will be handling are co-linear groups. As the primitives are defined as local contour descriptors, it follows that continuous groups of locally co-linear primitives describe those precise contours. Therefore we defined a local version of the co-linear relation, and a transitivity relation, such that if two primitives A and B are co-linear, and a third primitive C is co-linear to B yet not to A, then we consider that (A,B,C) is a contour.

The advantage of considering contours instead of individual primitives is that contours are expected to be extracted in a robust manner from the visual signal: as long as an object is visible its contours will be extracted, and assuming that the object is subject to only a moderate motion, the contour aspect will change smoothly over time (due to perspective change). On the other hand, the primitives that constitute a contour will vary due to local signal noise.

As an example of higher visual Gestalt we will present the case of parallelism. Two primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ are called parallel iff:

$$1 - \frac{2}{\pi}\text{acos}(|\Theta_i \cdot \Theta_j|) < \tau_{\|} \tag{4}$$

where Θ_i is the orientation vector of the primitive $\mathbf{\Pi}_i$ and $\tau_{\|}$ is the parallelism tolerance. By extension we will consider that two contours G_1 and G_2 are parallel iff:

$$\begin{array}{c} \forall\, \mathbf{\Pi}_i \in G_1, \mathbf{\Pi}_j = \arg\min_{\mathbf{\Pi}_k \in G_2}(d_E(\boldsymbol{M}_i, \boldsymbol{M}_k)), \\ \left\{ \begin{array}{c} 1 - \frac{2}{\pi}\text{acos}(|\Theta_i \cdot \Theta_j|) < \tau_{\|} \\ |d_E(\boldsymbol{M}_i, \boldsymbol{M}_k) - d_E(G_1, G_2)| < \tau_E \end{array} \right. \end{array} \tag{5}$$

where $\boldsymbol{M}_i$ is the position of the primitive $\mathbf{\Pi}_i$, d_E is the Euclidian distance, $d_E(G_1,G_2)$ is the min distance between the two contours, and τ_E is the tolerance. Note that the second line is the global constraint for parallelism, whereas the first line is the local one. The first constraint statistically weakens with larger contours whereas the second one strengthens. Note that this interpretation of parallelism can capture curved contours that are equidistant in all points.

In Figure 3 an image from a driving scenario is shown. 3(b) shows the contours extracted, and 3(c) the Visual Gestalts obtained after merging parallel contours together. This process is described in detail in [J].

3.2 Surface prediction at homog. surfaces

It is known that it is hard or impossible to extract 3D information at image areas that are weakly structured or completely homogeneous by methods based on correspondences. However, it is possible to predict a depth for these areas by making use of contour information [D]. We have formalized the visual analog of an edge primitive for a homogenuous image patch that we call a 'mono'.

$$\mathbf{\Pi}^{\mathrm{m}} = (\boldsymbol{M}, \vec{\mathbf{n}}, \mathbf{c}), \tag{6}$$

where $\boldsymbol{M}$ is the position of the center of the mono; $\vec{\mathbf{n}}$ is the normal of the plane that defines the local surface patch represented by the mono; and, $\mathbf{c}$ is the color representation of the mono. There is a strong relation between edge primitives and monos in the sense that it is possible to predict depth information at homogeneous surfaces through the bounding edges (see [E]).

Based on these observations, we have developed a voting model which predicts the depth at mono primitives *from* the depth of the bounding edge primitives, utilizing coplanarity, co-colority and co-linearity relations. Figure 4(f) shows the bounding edges for a mono. Each pair of edges from this list of bounding edges vote for a depth and a surface normal for the mono. Our model combines these votes and assigns the results of this combination as the depth and the surface normal of the mono.

Due to outliers in the stereo (see Figure 4(d)), the predictions have also outliers which we remove by combining the depth predictions in image areas (see Figure 4(e)). This improvement assumes that the monos in an image area are part of the same surface. In other words, for each image area, a generic surface model which allows spherical, quadratic, hyperbolic as well as planar shapes is fitted to the mono predictions in that area. The results of such predictions are shown in Figure 4(g). The results show that in spite of outliers in the stereo data (shown in Figure 4(d)), our model is able to predict the surfaces at homogeneous image patches.

4. Multi-sensorial primitives

In our robot environment we can relate monos (and eventually also edge primitives) to tactile information. We are currently investigating a specific type of tactile sensor (described in Section 4.1 and [F]). This sensor delivers partly complementary as well as comparable information to the Monos introduced in Section 3.2. This information can then be combined to a multi-sensorial primitive.

4.1 Tactile sensor

The sensor we are using here has been introduced in WP1 and is also described in Deliverable D1.2.

The purpose of the tactile sensor system is the support of haptic exploration and controlled grasping skills for the robot. The following requirements arise from this application:

High sensitivity and wide measurement range: Detection of slight contacts of a few gram weight equivalent should be possible, but the sensor must also not be overdriven when moving or lifting weights in the range of $1-2Kg$.

Response dynamics: The sensor signal should have a rise time below $20ms$ to allow the implementation of controlled grasping.

Reliability: A strong demand for the choice of the sensor is a proven sensor technology which affords little maintenance and has a sufficient life time.

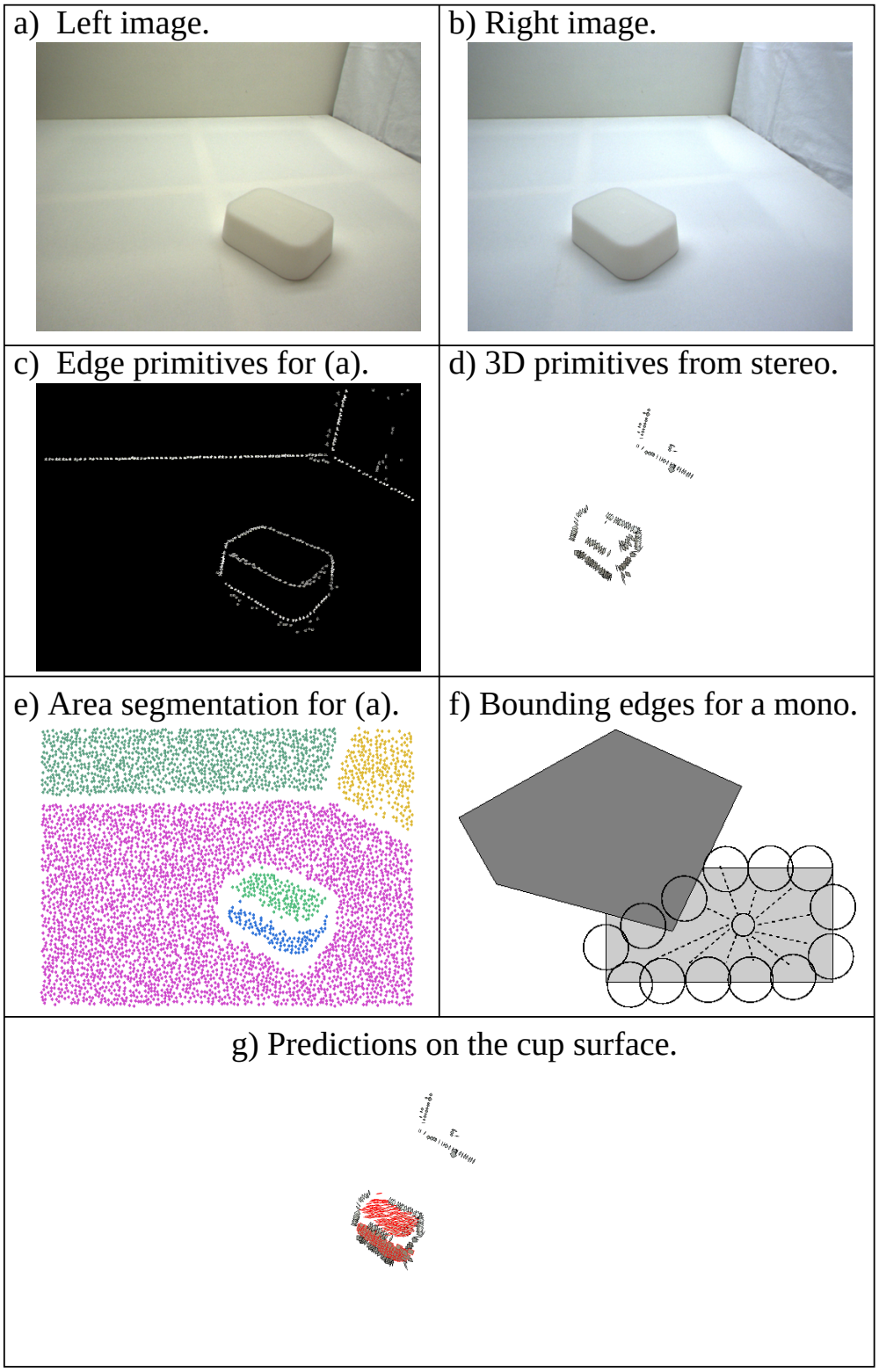


Figure 4: Illustration of depth prediction on an example scene. **(a,b)** Input stereo pair. **(c)** 2D edge primitives extracted from (a). **(d)** 3D edge primitives computed using stereo. **(e)** Segmentation of 2D monos in (a) into areas using co-colority. **(f)** Bounding edges for a mono. **(g)** 3D monos predicted from (d) using area information in (e).

Size and ease of integration: The sensor device should be small enough to fit into the phalanxes of the Karlsruhe Robot Hand [17], which is of the size of the human hand.

Electrical interface and measurement electronics: The sensor should provide an electrical interface with low cable count and that is not sensitive towards moderate electrical interference. The measurement electronics must be small in size and should offer a standard PC communication interface like RS232 or USB.

Beside these basic demands a further strong requirement for the tactile sensors is the capability to determine the contact normal force vector (CNFV) which allows for dextrous manipulation and reactive grasping with several common control algorithms [9, 4].

Based on the demands defined above we decided to investigate MicroJoysticks (see Figure 5) and MicroNavs (see Figure 6) in terms of their potential to extract relevant tactile information. Figure 5 shows a jaw gripper equipped with MicroJoystick. Here four devices were soldered to a printed cirtuit board (PCB) to investigate the feasibility of a matrix sensor field. Each device consists of four smaller force sensors located in four directions (north, south, east and west). Each device thus gives four force readings making it possible not only to measure the magnitude of the applied force but also the direction.

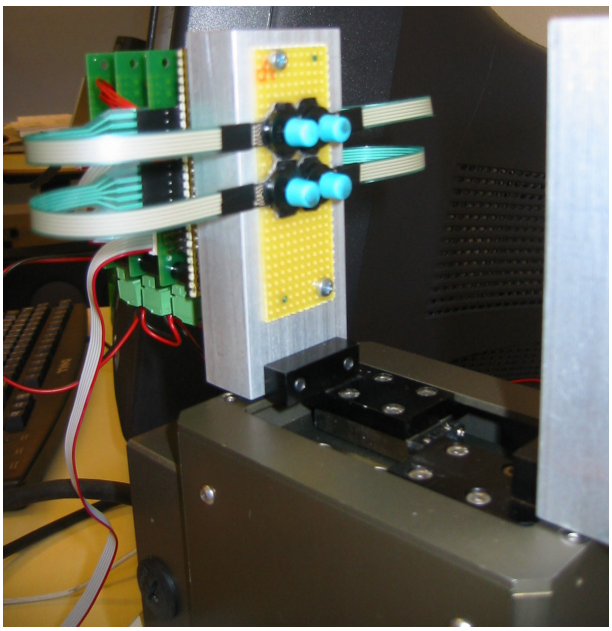


Figure 5: Four *MicroJoystick* devices mounted to a jaw gripper.

The data acquisition circuit is attached to the backside of one jaw, also the typical flex board cables as needed for the *MicroJoystick* device [1] are visible. This setup is used as a demonstrator for investigating the characteristics of the *MicroJoystick* cursor navigation sensor in tactile exploration. As mentioned before the bulky geometry of the sensors actuator cap and the connection wires make this setup sensitive towards mechanical damage. Also, the grippers contact area is reduced to the actuator caps front surface which is not suitable for clamping objects.

In a second approach cursor navigation sensors with silicone actuator caps were integrated into a humanoid robot hand [17]. Figure 6 shows the sensor element integrated into the thumb tip of the humanoid robot hand. The active sensor area is covered with a thin layer of silicone that was adapted to the shape of the underlying silicone finger tip. This silicone cap naturally flows to a spherical shape which results in a proper actuator for this sensor. An advantage of this design is that the finger surface area is not affected by the integration of the sensor and the stable mechanical design of the finger tips can be maintained.

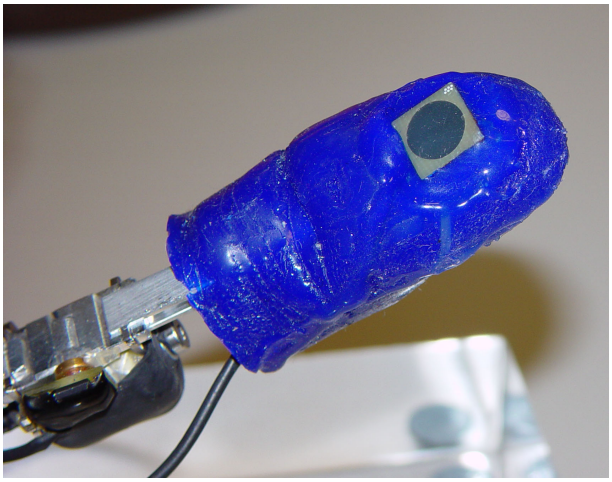
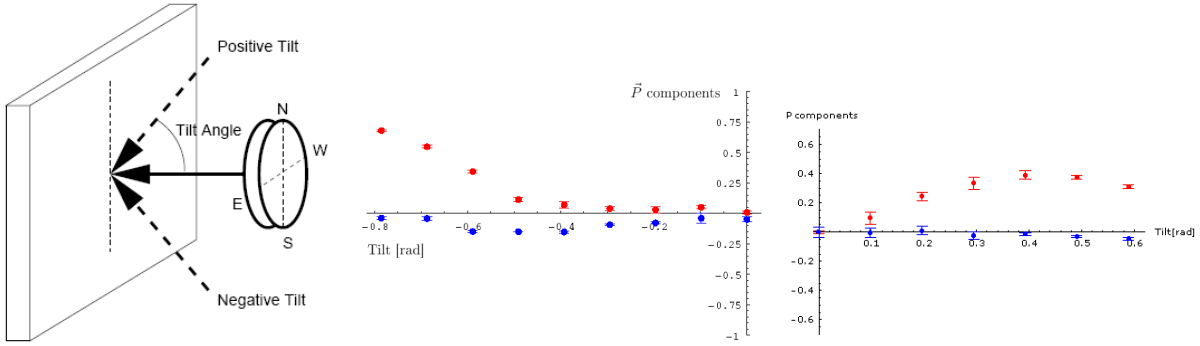


Figure 6: Integration of *MicroNav 360* sensor device .



(a) Definition of tilt angle (b) Results using the *MicroJoystick* sensor (c) Results using the *MicroNav* sensor

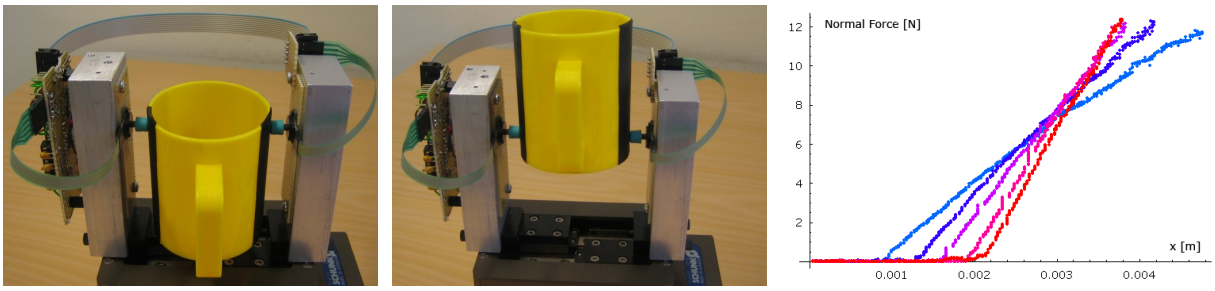
Figure 7: Surface Normal Experiment

4.1.1 Experiments for the extraction of surface normals, elasticity and weight

A series of experiments were carried out to investigate whether the tactile sensors can be used to measure and extract object and surface properties. In these experiments we explored the ability of the *MicroJoystick* and *MicroNav* sensors to measure the normal of a surface by touching it with a single sensor mounted on the tip of a robot finger. Additionally an object was grasped using a parallel gripper with one sensor mounted on each finger to explore whether the sensor setup is able to measure the elasticity of a surface and the weight of an object.

Surface Normal: The orientation of the surface is defined relative to the sensor using two angles instead of using the surface normal vector. These angles are the roll angle β and the tilt angle α. The tilt is the angle at which the sensor is tilted relative to the surface normal (see Figure 7(a)). The roll angle defines in which direction the sensor is tilted. The reason for this definition is that the ability of the sensor to measure each of these angles could be investigated separately.

During the experiment the output of the sensor was converted into a two dimensional vector ($\vec{P}$). See [F] for details. This vector describes the difference in the readings in the two axes of the sensor respectively, relative to the total force measured by the sensor. This vector thus only depends on the direction of the force vector applied to the sensor, and is in theory independent of the magnitude. The output of the *MicroJoystick* sensor was recorded for a series of different tilt angles in the range $-\frac{\pi}{4} \leq \alpha \leq 0$. The measured values of $\vec{P}$ for each of the tilt angles can be seen in Figure 7(b). The output of the *MicroNav* sensor was recorded for



(a) Top (b) Bottom (c) Results

Figure 8: Softness Experiment Setup

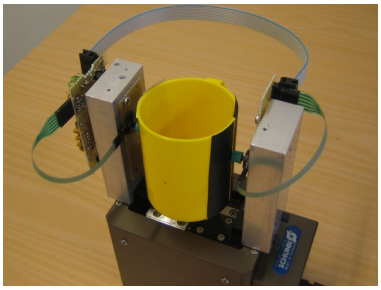
a series of different tilt angles in the range $0 \leq \alpha \leq \frac{\pi}{4}$. The measured values of $\vec{P}$ for each of the tilt angles can be seen in Figure 7(c). The red markings show the values corresponding to the axis in which the sensor was tilted, and should thus give a readout depending on the tilt angle. The blue markings show the values corresponding to the other axis, so these values are expected to be close to zero.

These experiments show that the *MicroJoystick* is unable to detect a tilt angle lower than about 22°. For a higher tilt angle the readings from the sensor grows rapidly. On the other hand the readings from the *MicroNav* sensor seems to have a linear relationship with the tilt angle although this relationship does not seem to hold for angles higher than about 22°. Additionally the experiments showed that the *MicroNav* sensor could also be used to measure the roll angle. It would thus be possible to measure the surface direction using the *MicroNav* sensor.

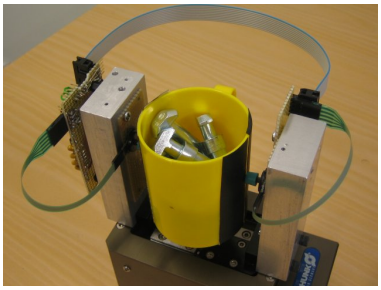
Elasticity: The elasticity of an object or a surface is the ability to deform when a force is applied to a contact point, for example during a grasp. The elasticity is a useful property to be known in grasping, since it makes it possible to predict in what way the object will deform during a grasp. It as also a relavant object property. A high elasticity makes an object for example useless to be used in a hammer like way. The ability of the sensor to measure the elasticity of an object was explored using a two sensor setup mounted on a parallel gripper. The gripper would close around a plastic cup with the two sensors as the only contact points. When the cup was grasped in the top it would deform into a oval shape, since it is more flexible at this point (see Figure 8(a)). In the lower part of the cup the shape was stabilized by the bottom of the cup, and would not easily deform (see Figure 8(b)). The parallel gripper was closed slowly with a constant velocity and stopped when a certain maximum force was reached. The diameter of the cup at the top was a little larger than at the bottom. It was measured to $59mm$ at the top versus $57.5mm$ at the bottom. The experiment was repeated 5 times with different contact locations. The force measured by one of the sensors as the gripper was closing can bee seen in Figure 8(c). The light blue graph shows the result when the contact point was at the top of the cup, and the red graph shows for the bottom of the cup. The rest of the graphs show the contact points in between these two.

It can clearly be seen that the force is growing slowly when grasping at a soft location, and growing fast when grasping at a hard location. The sensor is also able to detect the different diameter of the cup, depending on the grasping point. When grasping at the top the sensor measures a contact after $\approx 1mm$ of movement, and when grasping at the bottom it measures a contact after $\approx 2mm$ of movement because of the smaller diameter.

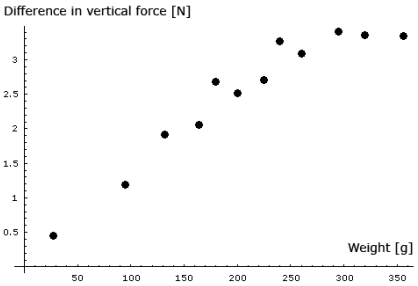
Weight: The sensors are not expected to be able to measure the precise weight of an object, but we find it useful to investigate whether they are able to give an indication of the weight of a grasped object to attach properties such as 'fullness' or 'emptyness'. For this experiment the same sensor setup was used, and the sensor readings were recorded when grasped objects with constant shape but different weights. The gravitational force would exert a downward force on the object, so the weight was expected to be measurable



(a) Empty



(b) Full



(c) Results

Figure 9: Weight Experiment Setup

in the force readings of the vertical axis of the sensor. The plastic cup was grasped one time where it was empty (see Figure 9(a)) and several times filled with different amounts of metal objects (see Figure 9(b)).

The result can be seen in Figure 9(c). The y-axis shows the average of the force difference measured in the vertical direction of the two sensors This seems to depends linear on the weight of the object except for weights higher than $\approx 300g$, where it seems to approach a limit.

4.2 Tactile Monos and multi-sensorial Primitves

The experiments in Section 4.1 showed that besides position information (which is trivial knowing the position of the robot end effector) we can extract surface normal information using the tactile sensor. Furthermore, also information about the local elasticity and the weight can be extracted.

Hence, we can define a 3D tactile primitive $\mathbf{\Pi}^{\mathrm{t}}$ by the 3D postion, the surface normal $\vec{\mathbf{n}}$ as well as by indicators for properties such as elasticity e and weight w:

$$\mathbf{\Pi}^{\mathrm{t}} = (\mathbf{X}, \vec{\mathbf{n}}, e, w).$$

We can now combine visual and tactile information in one multi-sensorial primitive $\mathbf{\Pi}^{\mathrm{ms}}$ by

$$\mathbf{\Pi}^{\mathrm{ms}} = (\mathbf{\Pi}^{\mathrm{m}}, \mathbf{\Pi}^{\mathrm{t}}).$$

Note that we are still working on the optimal sensor shape and sensor arrangement such that the definition of these measures is still ongoing. For details, see [F].

4.3 Relation of tactile monos to visually predicted Monos

The definition of the tactile primitive resembles the definition of the visual mono except that it does not cover colour information which is a purely visual attribute. However, there are two differences between the two kinds of primitives caused by two differences of the underlying senses:

- Visual monos can be computed globally (i.e., over all surfaces in the scene) while a tactile primitive can only extract information locally (i.e., where there is direct touch information).
- The information of the tactile primitive can be seen as more reliable than the visual primitive. While, in particular at the current stage of processing of visual primitives, a good amount of monos become predicted wrongly since many co-planar structures might not be caused by surfaces, a touch is a very

reliable information that there must be somethung there. Hence, touch can be used to verify and correct visual predictions.

4.4 Example 2: Interaction of vision and touch for shape extraction

To combine the higher reliability of the haptical monos with the globally available visual monos the visual features can guide the haptical exploration. In the following, we describe an experiement where the *Micro-Joytick* sensor has been used to verify whether a surface predicted by the vision system actually exists. The scene chosen for verification consists of a closed white box placed on a black surface (see figure 10(a)). The vision system predicts three possible surfaces in the scene (see figure 10(b)).

Each of these three detected surfaces consist of a group of visual mono primitives each describing among other things the position of a point on the surface and the surface normal in that point. This information can be used to create a trajectory where the sensor is moved through a point on the predicted surface in a linear movement parallel to the surface normal. Does the sensor come into contact with the expected surface the surface normal can be extracted and the surface is validated.

Figure 10(c) shows a situation where a visual mono on top of the box has been chosen. The robot moves the gripper in position above the surface, and then does a straight line movement through the surface. Since a contact is detected by the sensor (see figure 10(d)) a haptic mono primitive is added at the point of contact. The top surface of the box was verified three times, resulting in three haptic mono primitives (figure 10(e)).

The vision system also predicted a surface located between the edge of the box and the edge of the black surface. To verify this surface a visual mono was chosen and the robot did the same straight line movement though the predicted surface. Since no surface exist in the point the robot moves through the predicted surface without detecting a contact and stops (see figure 10(f)).

Since the visual mono primitives on the same surface are grouped together it is possible to validate or disprove all the visual monos of a surface in one step.

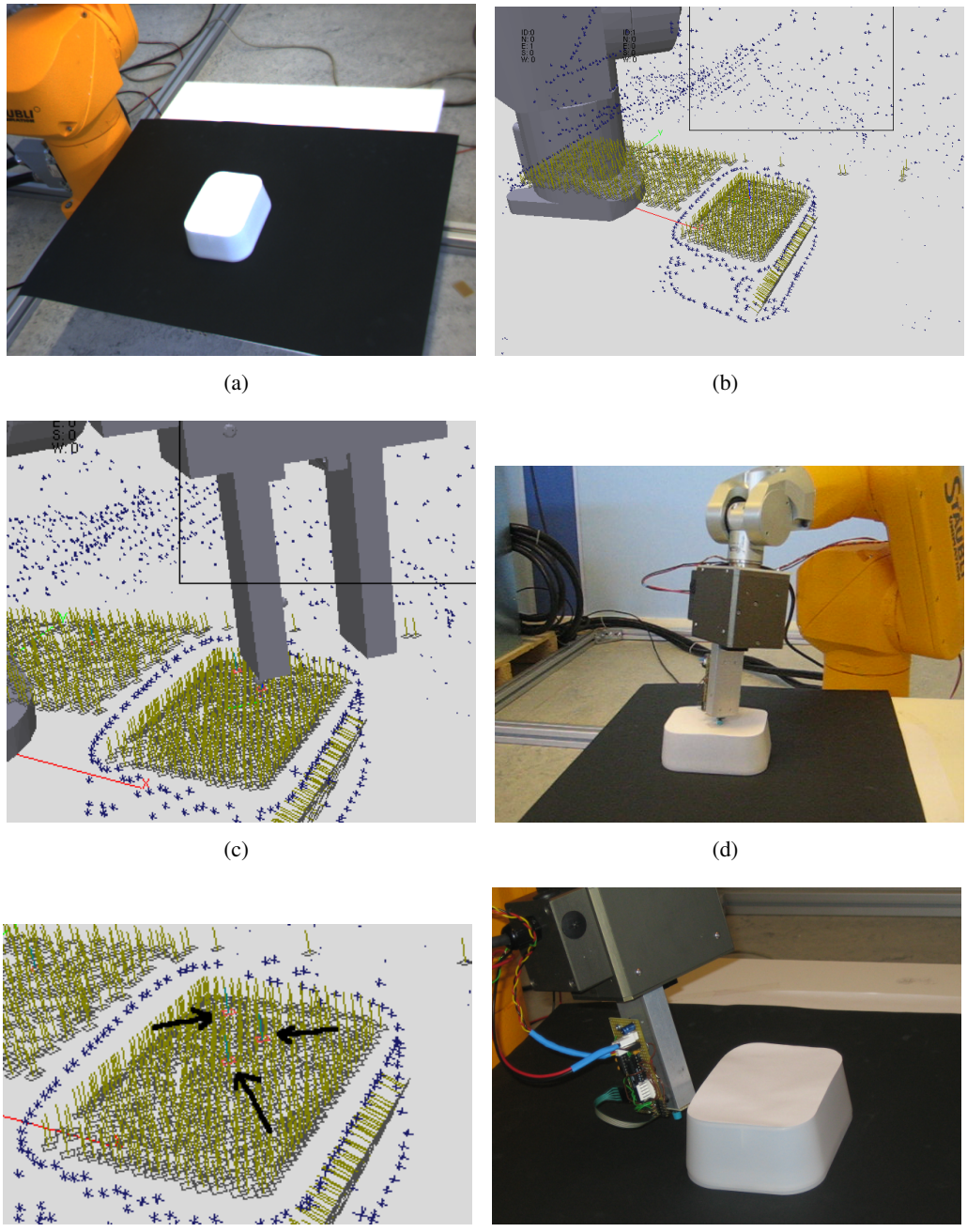
4.5 Current and future research

We are currently working on the extraction of richer tactile information to include these in the tactile primitives:

- Grouping multiple micro-joysticks together, we aim at the extraction of torque which is important for, e.g., the evaluation of grasp stability (see, e.g., [3]) or the estimation of weight.
- Extracting tactile texture from dynamic information gained by small explorative movements over objects.
- The evaluation of different sensor forms in respect to measuremet quality for different tasks.

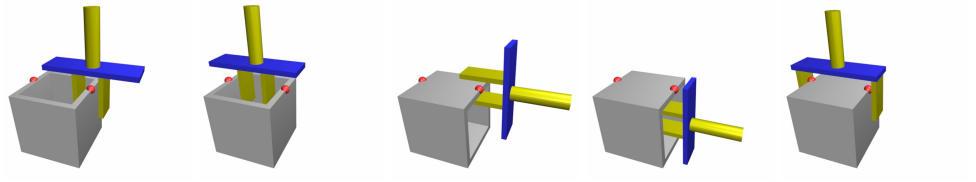
5. Example 3: Haptically evaluated vision-based grasping

One of the most basic interactions that can occur between a robot and an object is for the robot to push the object, i.e. to simply make a physical contact. Already at this stage, the robot should be able to form two categories: physical and non-physical objects, where a physical object is categorized by the fact that interaction forces occur. A higher level interaction between the robot and an object would exist if the robot was able to *grasp* the object. In this case, the robot would gain actual physical control over the object and having the possibility to perform controlled actions on it, such as examining it from other angles,



(a) (b) (c) (d) (e) (f)

Figure 10: Surface Verification Experiment. **(a)** Setup of the scene. **(b)** Three predicted surfaces. **(c)** The robot moving in position to verify the surface on the box. **(d)** The sensor in contact with the surface on the box. **(e)** Three detected haptic primitives shown as small red squares. A green line marking the surface normal. **(f)** The robot moving through the wrongly predicted surface without detecting a contact.



(a) EGA1 (b) EGA2 (c) EGA3 (d) EGA4 (e) EGA5

Figure 11: Elementary grasping actions, EGAs. The small red balls represent co-planar visual primitives.

weighing it, placing it etc. Information obtained during this interaction can then be used to update the robots representations about objects and the world. Furthermore, the successfully performed grasps can be used as ground truth for future grasp refinement [7].

Here, we are interested in investigating an initial "reflex-like" grasping strategy that will form a basis for a cognitive robot system that, at the first stage, acquires knowledge of objects and object categories and is able to further refine its grasping behavior by incorporating the gained object knowledge [2]. The grasping strategy does not require *a-priori* object knowledge, and it can be adopted for a large class of objects. Note that this early OAC that is actually applied to arrive at the concept of an object (see D8.1.1).

The proposed reflex-like grasping strategy is based on second order relations of the multi-modal visual primitives that represent object's geometric information, e.g. 3D pose (position and orientation) as well as its appearance information. Co-planar tuples of the spatial primitives allow for the definition of a plane that can be associated to a grasp hypothesis (see Figure 11). In addition, these local descriptors are part of semi-global co-linear groups [K]. Furthermore, the color information (by defining co-colority in addition to co-planarity of primitive pairs) can be used to further improve the definition of grasp hypotheses. Hence, we employ the structural richness of the descriptors in terms of their geometry and appearance as well as the structural relations co-linearity, co-planarity and co-colority to derive grasping options from a stereo image. Figure 12 shows some of the generated grasping hypotheses. This work is described in detail in [A].

Performing a grasping hypothesis, closing the gripper and then evaluating the distance between the two fingers that is given as haptic information from the gripper, we have a haptic criterion for a successful grasp. Hence, although initially a purely visual Gestalt, the Gestalt becomes multi-sensory taking proprioceptive information (robot motion and finger distance).

5.1 Current and future research: Refinement of grasps

The reflex-like elementary grasping actions are hard-wired and based on limited, task-independent, bottom-up information extracted from visual input. Thus, they will succeed only up to a certain level of reliability. However, they constitute a crucial initial step for bootstrapping a grasp learning procedure that produces *associative* grasping actions. We will develop learning procedures that permit a robot to learn robust, object-specific grasp behaviors by trial and error, associating object-specific, highly predictive visual features with kinematic parameters that lead to successful grasps with high probability [5]. For this to work, the following key capabilities are required of the robot:

1. to extract sufficiently predictive and distinctive visual features of the object,
2. to try out a variety of different grasps,
3. to evaluate the success of a grasp after it has been applied.

There are different ways of evaluating graps (Capability 3), e.g. based on verifying physical control over the grasped object using vision, or based on haptic feedback [4].

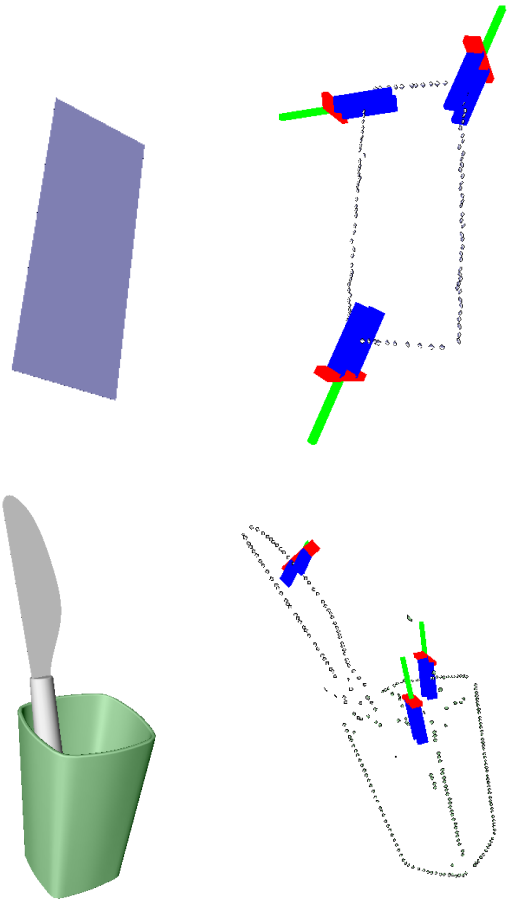


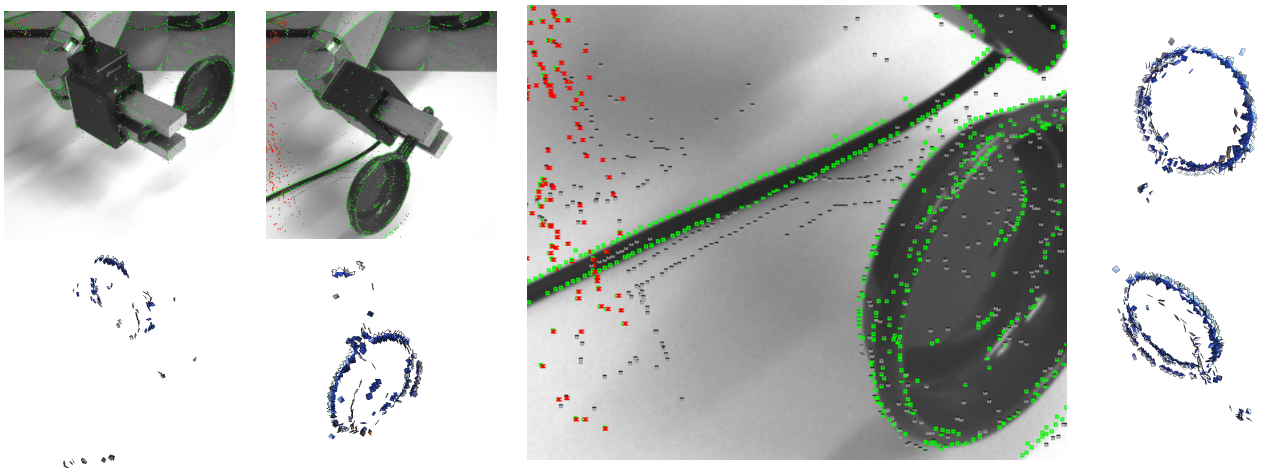
Figure 12: Two example scenes designed for testing and a selection of the generated actions.

For trial-and-error learning (Capability 2), it is essential that exploratory graps accidentally succeed sufficiently often to allow learning over a feasibly small number of trials. Given the multiple-dimensional and continuous parameter space of robotic manipulators, purely random exploration appears unrealistic. Rather, heuristics are required that bias exploration towards promising regions of the parameter space. The reflex-like grasping actions will serve this purpose; a success rate around 10% should be more than sufficient.

The learning procedure will identify successful grasp parameters and associate them with object appearance, such that after learning, static visual input of an object is sufficient to retrieve the most robust grasp parameters for this object. The success of this idea hinges on the ability to extract sufficiently distinctive and descriptive visual features of the set of objects of interest (Capability 1). To this end, we plan to build on our current work on learning probabilistic object representations in terms of local appearance and spatial relations that have already proven useful for 2D object detection and recognition [16, 15]. We have recently generalized this framework to 6D pose space, allowing visual primitives (such as second-order relations of multi-modal features) and effector pose to share the same representation, and have obtained first promising pilot results on pose estimation (which implicitly influences grasping kinematics).

6. Example 4: Birth of an object based on self-induced motion

If the motions of the objects within the scene are known, then the relation between features in two subsequent frames becomes deterministic (excluding the usual problems of occlusion, sampling, etc). This means that



(a) (b) (c) (d)

Figure 13: Birth of an object (a)–(b) top: 2D projection of the accumulated 3D representation and newly introduced primitives, bottom: accumulated 3D representation. (c) newly introduced and accumulated primitives in detailed. Note that, the primitives that are not updated are red and the ones that have low confidence are grey (d) final accumulated 3D representation from two different poses.

a structure (e.g. in our case a contour) that is present in one frame is guaranteed to be in the previous and next frames (provided it does not become occluded or goes out of the field of view of the camera), subject to a transformation that is fully determined by the motion: generally a change of position and orientation. If we assume that the motions are reasonably small compared to the frame-rate, then a contour will not appear or disappear unpredictably, but will have a life-span in the representation, between the moment it entered the field of view and the moment it leaves it (partial or complete occlusion may occur during some of the time-steps).

These prediction are relevant in different contexts

- **Establishment of objectness:** The objectness of a set of features is characterised by the fact that they all move according to the robot motion. This property is discussed in the context of a grounded AI planning system in [8] and Deliverable 8.1.1 where it is also shwon how a first affordance can be assigned to the object.
- **Segmentation:** The system segments the object by its predicted motion from the other parts of the scene.
- **Disambiguation:** Ambiguous features can be characterised (and eliminated) by not moving according to the predictions (see Figure 13).
- **Learning an object model:** A full 3D model of the object can be extracted by merging different views created by the motion of the end effector.

As a result we end up with a mutli-sensory Gestalt consisting of a set of entities that belong to the same object that are related by proprioceptive information (birth of an object, see delieverbale 8.1.1), see Figure 13. The details of the algorithm are decribed in [H].

We applied the accumulation scheme to a variety of scenes where the robot arm manipulated several objects. The motion was a rotation of 5 degrees per frame. The accumulation process on one such object is illustrated in Fig. 13. The top row show the predictions at each frame. The bottom row, shows the 3D-primitives that were accumulated (frames 1 and 22). The object representation becomes fuller over time, whereas the primitives reconstructed from other parts of the scene are discarded. Figure 14 shows the accumulated

Figure 14: Objects and their related accumulated representation.

representation for various objects. The hole in the model corresponds to the part of the object occluded by the gripper. Accumulating the representation over several distinct grasps of the objects would yield a complete representation.

7. Links to other Workpackages

In this work the haptic sensor introduced in WP1 is evaluated in the context of exploration. The work presented here also closely relates to WP2 that has the objective to equip the existing robotic platforms with integrated sensorimotor capabilities, necessarry to explore the 'things' of interest. Different from the research currently pursuaded in WP2, WP4 does not consider the control of the oculomotor system. In addition, WP4 explores multi-sensorial primitives related to exploration of 'things' while WP2 assumes that we deal with 3D shape primitives or already known objects.

8. Publications arising from the Project

The attached publications and reports [A, B, C, D, E, F, G, H, I, J, K] and the publication [16] have been published with support of the project.

Attached Papers

[A] D. Aarno, J. Sommerfeld, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger. Early Reactive Grasping with Second Order 3D Feature Relations, journal = IEEE International Conference on Robotics and Automation (ICRA), Workshop: From features to actions - Unifying perspectives in computational and robot vision, year = 2007,.

[B] S. Kalkan, N. Pugeault, and N. Krüger. Perceptual operations and relations between 2d or 3d visual entities. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-3, 2007.

[C] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of local 3d structure in 2d images. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1114–1121, 2006.

[D] S. Kalkan, F. Wörgötter, and N. Krüger. Depth prediction at homogeneous image structures. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-2, 2007.

[E] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of second-order relations of 3d structures. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[F] M. Kjaergaard, Dirk Kraft, Alex Bierbaum, Tamim Asfour, Rüdiger Dillmann, and Norbert Krüger. Using tactile sensors for multisensorial scene explorations. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-5, 2007.

[G] N. Krüger, N. Pugeault, and F. Wörgötter. Multi-modal primitives: Local, condensed, and semantically rich visual descriptors and the formalization of contextual information. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-4, 2007.

[H] N. Pugeault, Emre Baseski, Dirk Kraft, F. Wörgötter, and N. Krüger. Extraction of multi–modal object representations in a robot vision system. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[I] N. Pugeault, F. Wörgötter, , and N. Krüger. Rigid body motion in an early cognitive vision framework. In *Proceedings of the IEEE Systems, Man and Cybernetics Society Conference on Advances in Cybernetic Systems*, 2006.

[J] N. Pugeault, F. Wörgötter, , and N. Krüger. Structural Visual Events. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-1, 2007.

[K] N. Pugeault, F. Wörgötter, and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proc. IEEE Workshop on Perceptual Organization in Computer Vision (in conjunction with CVPR'06)*, 2006.

References

[1] Interlink electronics. http://www.interlinkelec.com.

[2] Pedram Azad, Tamim Asfour, and Ruediger Dillmann. Combining Appearance-based and Model-based Methods for Real-time Object Recognition and 6D Localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[3] A. Bicchi, J.K. Salisbury, and D.L. Brock. Contact sensing from force measurements. *The International Journal of Robotics Research*, 12(3), 1993.

[4] Jefferson A. Coelho, Jr. and Roderic A. Grupen. A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 14(7):545–557, 1997.

[5] Jefferson A. Coelho, Jr., Justus H. Piater, and Roderic A. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. *Robotics and Autonomous Systems*, 37(2–3):195–218, 2001.

[6] James H. Elder. Are edges incomplete ? *International Journal of Computer Vision*, 34:97–122, 1999.

[7] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini. Learning About Objects Through Action - Initial Steps Towards Artificial Cognition. In *IEEE Int. Conf on Robotics and Automation*, pages 3140–3145, 2003.

[8] Ch. Geib, K. Mourao, R. Petrick, N. Pugeault, M. Steedman, N. Krüger, and F. Wörgötter. Object action complexes as an interface for planning and robot control. *Workshop 'Toward Cognitive Humanoid Robots' at IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, 2006.

[9] R. D. Howe. Tactile sensing and control of robotic manipulation. *Journal of Advanced Robotics*, 8(3):245–261, 1994.

[10] http://www.pspc.dibe.unige.it/ drivsco/, editor. *DRIVSCO: Learning to Emulate Perception-Action Cycles in a Driving School Scenario (FP6-IST-FET, contract 016276-2)*. 2006-2009.

[11] K. Koffka. *Principles of Gestalt Psychology*. Lund Humphries, London, 1935.

[12] K. Köhler. *Gestalt Psychology: An introduction to new concepts in psychology*. New York: Liveright, 1947.

[13] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 1(5):417–428, 2004.

[14] N. Krüger and F. Wörgötter. Multi-modal primitives as functional models of hyper-columns and their use for contextual integration. *Proceedings of the 1st International Symposium on Brain, Vision and Artificial Intelligence 19-21 October, 2005, Naples, Italy, Lecture Notes in Computer Science, Springer, LNCS 3704*, pages 157–156, 2005.

[15] Fabien Scalzo and Justus H. Piater. Statistical learning of visual feature hierarchies. In *IEEE Workshop on Learning in Computer Vision and Pattern Recognition*, volume 3, pages 44–44, 2005.

[16] Fabien Scalzo and Justus H. Piater. Unsupervised learning of dense hierarchical appearance representations. In *International Conference on Pattern Recognition*, 2006.

[17] S. Schulz, C. Pylatiuk, and G. Bretthauer. A new ultralight anthropomorphic hand. In *IEEE International Conference on Robotics and Automation (ICRA)*.

In IEEE International Conference on Robotics and Automation, ICRA'07, *Workshop: "From features to actions - Unifying perspectives in computational and robot vision"*, April, 2007, Rome, Italy (to be presented)

Early Reactive Grasping with Second Order 3D Feature Relations

Daniel Aarno, Johan Sommerfeld, Danica Kragic
Royal Institute of Technology, Sweden
{bishop, johansom, dani}@kth.se

Nicolas Pugeault
University of Edinburgh, UK
npugeaul@inf.ed.ac.uk

Sinan Kalkan, Florentin Wörgötter
University of Göttingen, Germany
{sinan, worgott}@bccn-goettingen.de

Dirk Kraft, Norbert Krüger
Sydansk University and Aalborg University, Denmark
{norbert, kraft}@mip.sdu.dk

***Abstract*— One of the main challenges in the field of robotics is to make robots ubiquitous. To intelligently interact with the world, such robots need to understand the environment and situations around them and react appropriately, they need context-awareness. But how to equip robots with capabilities of gathering and interpreting the necessary information for novel tasks through interaction with the environment and by providing some minimal knowledge in advance? This has been a longterm question and one of the main drives in the field of cognitive system development.**

The main idea behind the work presented in this paper is that the robot should, like a human infant, learn about objects by interacting with them, forming representations of the objects and their categories that are grounded in its embodiment. For this purpose, we study an early learning of object grasping process where the agent, based on a set of innate reflexes and knowledge about its embodiment. We stress out that this is not the work on grasping, it is a system that interacts with the environment based on relations of 3D visual features generated trough a stereo vision system. We show how geometry, appearance and spatial relations between the features can guide early reactive grasping which can later on be used in a more purposive manner when interacting with the environment.

I. Introduction

For a robot that has to perform tasks in a human environment, it is necessary to be able to learn about objects and object categories. It has been recognized recently that grounding in the embodiment of a robot, as-well as continuous learning is required to facilitate learning of objects and object categories [1], [2]. The idea is that robots will not be able to form useful categories or object representations by only being a passive observer of its environment. Rather a robot should, like a human infant, learn about objects by interacting with them, forming representations of the objects and their categories that are grounded in its embodiment.

Central to the approach are three almost axiomatic assumptions, which are strongly correlated. These also represent the building blocks of our approach toward creating a cognitive artificial agent:

- Objects and Actions are inseparably intertwined; Entities ("things") in the world of a robot (or human) will only become semantically useful "objects" through the action that the agent can/will perform on them. This forms so-called Object-Action Complexes (named OACs) which are the building blocks of cognition.
- Cognition is based on recurrent processes involving nested feedback loops operating on, contextualizing and reinterpreting object-action complexes. This is done through actively closing the perception-action cycle.
- A unified measure of success and progress can be obtained through minimization of contingencies which an artificial cognitive system experiences while interacting with the environment or other agents, given the drives of the system.

To demonstrate the feasibility of our approach, we aim at building a robot system that step by step develop increasingly advanced cognitive capabilities. In this paper, we demonstrate our initial efforts towards this goal by designing a scenario for manipulation and grasping of objects.

One of the most basic interactions that can occur between a robot and an object is for the robot to push the object, i.e. to simply make a physical contact. Already at this stage, the robot should be able to form two categories: physical and non-physical objects, where a physical object is categorized by the fact that interaction forces occur. A higher level interaction between the robot and an object would exist if the robot was able to *grasp* the object. In this case, the robot would gain actual physical control over the object and having the possibility to perform controlled actions on it, such as examining it from other angles, weighing it, placing it etc. Information obtained during this interaction can then be used to update the robots representations about objects and the world. Furthermore, the successfully performed grasps can be used as ground truth for future grasp refinement, [2].

In this paper, we are interested in investigating an initial "reflex-like" grasping strategy that will form a basis for

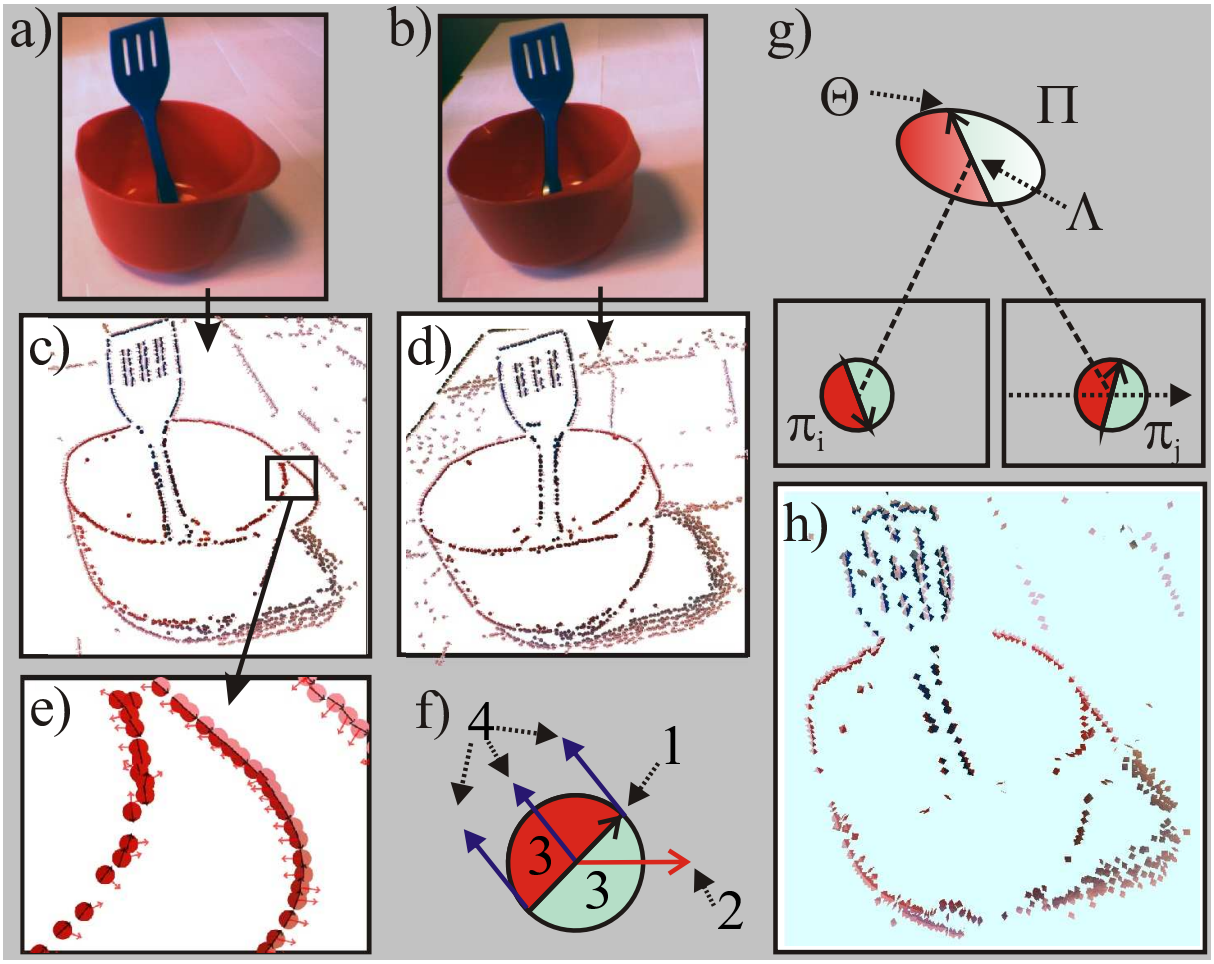


Fig. 1. Illustration of the vision module. a) and b) shows the images captured by the left and right cameras (respectively); c) and d) show the primitives extracted from these two images; in e) a detail of the primitive extraction is shown; f) illustrates the schematic representation of a primitive, where 1. represents the orientation, 2. the phase, 3. the color and 4. the optical flow. g) from a stereo–pair of primitives (π_i, π_j) we reconstruct a 3D primitive Π, with a position in space Λ and an orientation Θ; h) shows the resulting 3D primitives reconstructed for this scenario.

a cognitive robot system that, at the first stage, acquires knowledge of objects and object categories and is able to further refine its grasping behavior by incorporating the gained object knowledge, [3]. The grasping strategy does not require *a-priori* object knowledge, and it can be adopted for a large class of objects. The proposed reflex-like grasping strategy is based on second order relations of multi-modal visual features descriptors, called *spatial primitives*, that represent object's geometric information, e.g. 3D pose (position and orientation) as well as its appearance information, e.g. color and contrast transition etc. [4], see Fig. 1. Co–planar tuples of the spatial primitives allow for the definition of a plane that can be associated to a grasp hypothesis. In addition, these local descriptors are part of semi-global collinear groups [5]. Furthermore, the color information (by defining co–colority in addition to co–planarity of primitive pairs) can be used to further improve the definition of grasp hypotheses. In this paper, we employ the structural richness of the descriptors in terms of their geometry and appearance as well as the structural relations co–linearity, co–planarity and co–colority to derive a set of grasping options from a stereo image.

We note that the purpose of this work is not to develop yet another grasping strategy for a specific setting, but rather to provide low-level grasping reflexes that can be used to generate successful grasps on arbitrary objects. These grasping reflexes are part of a larger framework on cognitive robotics where a robot is equipped only with a set of innate grasps which are used to develop more complex object manipulation abilities through interaction and reinforcement so that 1) more complex feature relations become associated to more precise and successful grasps, and 2) object knowledge becomes acquired and used to further refine the grasping process. We also have to stress out that no scene segmentation is performed, since the system does not even have a concept of an object to start with. In short, the contributions of our work are the generation of a set of grasp suggestions on unknown objects based on visual feedback, grouping of visual primitives for decreasing the size of the grasps and evaluation of grasps using the GraspIt! environment, [6].

In this work, "kitchen-type" objects such as cups, glasses, bowls and various kitchen utensils are considered. However,

our algorithm is not designed for specific object classes but can be applied for any rigid object that can be described by edge–like structures.

This paper is organized as follows. In Section II, we shortly review the related work and in Section III give a general overview of the system. Details about extraction of spatial primitives are presented in Section IV and elementary grasping actions defined in Section V. Results of the experimental evaluation are summarized in Section VI and plans for future research outlined in Section VII.

II. Related Work

The idea to learn or refine grasping strategies is not new. Kamon *et al.* combined heuristic methods with learning algorithms to learn how to select good grasps [7]. Rössler *et al.* used two levels of learners to learn local and global grasp criteria [8], where the local learner learns about the local structure of an object and the global learner learns which of the possible local grasps are best given the object.

There has been a large amount of work presented in the area of robotic grasping during the last two decades [9]. However, much of this work has been dealing with analytical methods where the shape of the objects being grasped is known *a-priori*. This work, referred to as *analytical methods*, has focused primarily on computing grasp stability based on force and form-closure properties or contact-level grasps synthesis based on finding a fixed number of contact locations with no regard to hand geometry, [9],[10]. This problem is important and difficult mainly because of the high number of DOFs involved in grasping arbitrary objects with complex hands. Another important research area is grasp planning without detailed object models where sensor information such as computational vision is used to extract relevant features in order to compute suitable grasps, [11], [12], [13]. In this paper, we denote this approach as *sensor-driven*.

Related to our work, we have to mention systems that deal with automatic grasp synthesis and planning, [14],[15],[16],[17]. This work concentrates on automatic generation of stable grasps given assumptions about the shape of the object and robot hand kinematics. Example of assumptions may be that the full and exact pose of the object is known in combination with its (approximate) shape, [14]. Another common assumption is that the outer contour of the object can be extracted and a planar grasp applied, [16]. Taking into account both the hand kinematics as well as some *a-priori* knowledge about the feasible grasps has been acknowledged as a more flexible and natural approach towards automatic grasp planning [18],[14]. [18] studies methods for adapting a given prototype grasp of one object to another object. The method proposed in [14] presents a system for automatic grasp planning for a Barrett hand [19] by modeling an object as a set of shape primitives, such as spheres, cylinders, cones and boxes in a combination with a set of rules to generate a set of grasp starting positions and pregrasp shapes.

One difference between the analytical and sensor-driven approaches is that the former tend to use complex hands with many DOFs, while the latter use simple ones such as parallel yaw-grippers. One reason for this is that if the reconstruction of the object's shape is not very accurate, using a complex gripping device does not necessarily facilitate grasping performance. For sensor-driven approaches it is also very common to perform only planar grasps where all the contacts between the fingers and the object are confined to a plane. As an example, objects are placed on a table and grasped from above. This simplifies both the vision problem, since only the outer boundary of the object in the image plane has to be estimated, as well as the grasp planning by constraining the search space.

The main differences of our work compared to the above-mentioned work are the following:

- We rely on 3D information based on three dimensional primitives extracted online. This allows us to compute arbitrary grasping directions compared to only planar grasps considered in, e.g. [16].
- The structural richness of the primitives (geometric and appearance based information, collinear grouping) allows for an efficient reduction of grasping hypotheses while keeping relevant ones.
- Our system focuses on generating a certain percentage of successful grasps on arbitrary objects rather than high quality grasps on a constrained set of objects. We will show that with our representations we are able to extract a sufficient number of successful grasping options to be used as initiator of learning schemes aiming at more sophisticated grasping strategies.

III. System Overview

The work presented in this paper serves as a building block for the development of a cognitive robot system. The robot platform considered is comprised of a set of sensors and actuators. The minimum requirements necessary to realize the work presented in this paper is that the sensors are able to deliver a set of visual primitives (section IV) and the configuration of the actuators. The required actuator is a manipulator, comprised of a robotic arm and a gripper device. In this context the term sensor is not necessarily related to a real physical sensing device, but rather an abstract measurement delivered to the system, possibly after performing computations on data sampled from a physical sensor.

The complete system is outlined in Fig. 2. In this paper we are interested in developing grasping reflexes. A grasping reflex is triggered by the vision system. The vision system continuously computes the spatial primitives described in section IV which are feed as sensor input to the set of reflexes and to the cognitives system. If the grasping reflex has not been inhibited by the cognitive system and the sensor stimuli is strong enough, i.e. there are sufficiently many spatial primitives visible, the grasping reflex is performed. This reflex behavior computes a set of possible grasps and tries to perform them. Each grasp evaluated results in a reinforcement signal which can be used by the cognitive system to update its representation of the world. The following

two sections describe the spatial primitives and the rules for generating the grasping actions.

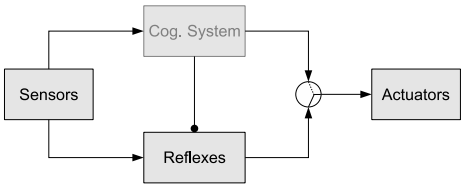


Fig. 2. System overview

IV. Spatial Primitives

The image processing used in this paper is based on multi-modal visual primitives [20], [4], [5]. First, 2D primitives are extracted sparsely at points of interest in the image (in this case contours) and encode the value of different visual operators (hereby referred to as *visual modalities*) such as local orientation, phase, color (on each side of the contour) and optical flow (see Fig. 1.d, 1.e and 1.f). In a second step, the 2D primitives become extended to the spatial primitives used in this work. After finding correspondences between primitives in the left and right image, we reconstruct a spatial primitive, (see Fig. 1.g) that has the following components, (for details see [21], [5]):

$$\Pi = \{\Lambda, \Theta, \Omega, (\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r)\},$$

where Λ is the 3D position; Θ is the 3D orientation; Ω is the phase (i.e., contrast transition); and, $(\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r)$ is the representation of the color of the spatial primitive, corresponding to the left ($\mathbf{c}_l$), the middle ($\mathbf{c}_m$) and the right side ($\mathbf{c}_r$).

The sparseness of the primitives allows to formulate three *relations* between primitives that are crucial in our context:

- *Co–planarity:*
  Two spatial primitives Π_i and Π_j are co–planar iff their orientation vectors lie on the same plane, i.e.:

  $$cop(\Pi_i, \Pi_j) = 1 - |\mathbf{proj}_{\Theta_j \times \mathbf{v}_{ij}}(\Theta_i \times \mathbf{v}_{ij})|,$$

  where $\mathbf{v}_{ij}$ is defined as the vector $(\Lambda_i - \Lambda_j)$, and $\mathbf{proj}_{\mathbf{u}}(\mathbf{a})$ is defined as:

  $$\mathbf{proj}_{\mathbf{u}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{u}}{\| \mathbf{u} \|^2}\mathbf{u}. \quad (1)$$

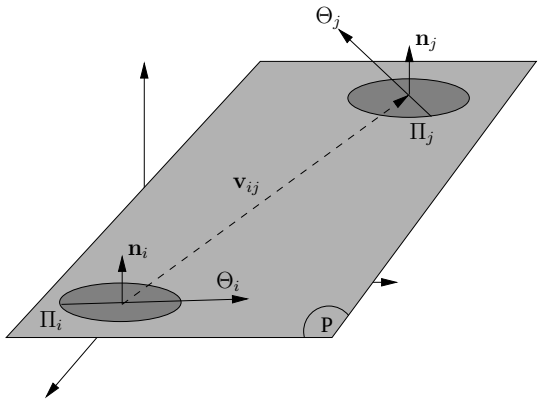
  The co–planarity relation is illustrated in Fig. 3(a).
- *Collinear grouping (i.e., collinearity):*
  Two spatial primitives Π_i and Π_j are collinear (i.e., part of the same group) iff they are part of the same contour. Due to uncertainty in 3D reconstruction process, in this work, the collinearity of two spatial primitives Π_i and Π_j is computed using their 2D projections π_i and π_j. We define the collinearity of two 2D primitives π_i and π_j as:

  $$col(\pi_i, \pi_j) = 1 - \left| sin\left(\frac{|\alpha_i| + |\alpha_j|}{2}\right)\right|,$$

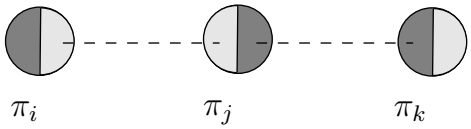
  where α_i and α_j are as shown in Fig. 3(c), see [5] for more details on collinearity.
- *Co–colority:* Two spatial primitives Π_i and Π_j are co–color iff their parts that face each other have the same color. In the same way as collinearity, co–colority of two spatial primitives Π_i and Π_j is computed using their 2D projections π_i and π_j. We define the co–colority of two 2D primitives π_i and π_j as:

  $$coc(\pi_i, \pi_j) = 1 - \mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j),$$

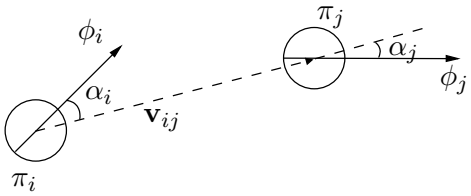
  where $\mathbf{c}_i$ and $\mathbf{c}_j$ are the RGB representation of the colors of the parts of the primitives π_i and π_j that face each other; and, $\mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j)$ is Euclidean distance between RGB values of the colors $\mathbf{c}_i$ and $\mathbf{c}_j$. In Fig. 3(b), a pair of co–color and not co–color primitives are shown.



(a) Co–planarity of two 3D primitives Π_i and Π_j.



(b) Co–colority of three 2D primitives π_i, π_j and π_k. In this case, π_i and π_j are cocolor, so are π_i and π_k; however, π_j and π_k are not cocolor.



(c) Collinearity of two 2D primitives π_i and π_j.

Fig. 3. Illustration of the relations between a pair of primitives.

Co–planarity in combination with the 3D position allows for the definition of a grasping pose; Collinearity and co–colority allows for the reduction of grasping hypotheses. The use of the relations in the grasping context is shown in Fig. 4.

V. Elementary Grasping Actions

Coplanar relationships between visual primitives suggests different graspable planes. Fig. 4 shows a set of spatial primitives on two different contours l_i and l_j with co–planarity, co–colority and collinearity relations.

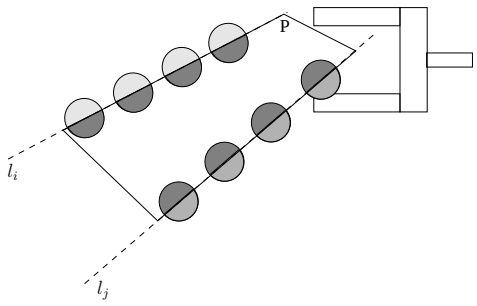


Fig. 4. A set of spatial primitives on two different contours l_i and l_j that have co–planarity, co–colority and collinearity relations; a plane P defined by the co–planarity of the spatial primitives and and an example grasp suggested by the plane.

Five elementary grasping actions (EGA) will be considered as shown in Fig. 5. EGA1 is a "pinch" grasp on a thin edge like structure with approach direction along the surface normal of the plane spanned by the primitives. EGA2 is an "inverted" grasp using the inside of two edges with approach along the surface normal. EGA3 is a "pinch" grasp on a single edge with approach direction perpendicular to the surface normal. EGA4 is similar to EGA2 but its approach direction is perpendicular to the surface normal. Also it tries to go in "below" one of the primitives. EGA5 is wide grasp making contact on two separate edges with approach direction along the surface normal.

The EGAs will be parameterized by their final pose (position and orientation) and the initial gripper configuration. For the simple parallel jaw gripper, an EGA will thus be defined by seven parameters: EGA$(x, y, z, \gamma, \beta, \alpha, \delta)$ where $\mathbf{p} = [x, y, z]$ is the position of the gripper "center" according to Fig. 6; $\gamma,\ \beta,\ \alpha$ are the roll, pitch and yaw angles of the vector $\mathbf{n}$; and δ is the gripper configuration, see Fig. 6. Note that the gripper "center" is placed in the "middle" of the gripper.

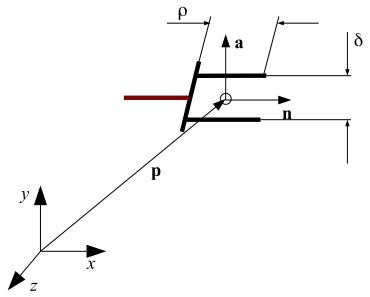


Fig. 6. Parameterization of EGAs.

The main motivation for choosing these grasps is that they represent the simplest possible two fingered grasps humans commonly use. The result of applying the EGAs can be evaluated to provide a reinforcement signal to the system. The number of possible outcomes of each of the EGAs are different and will be explained below.

For all of the EGAs the possibility of an *early failure* exists. That is, the EGA fails before reaching the target configuration. This will result in a reinforcement R_{fe}. Furthermore, it is possible for all EGAs to fail a grasping procedure.

For EGA1, EGA3 and EGA5, a failed grasp can be detected by the fact that the gripper is completely closed. This situation will result in a reinforcement R_{fl}.

For EGA1 and EGA3, the expected grasp is a pinch type grasp, i.e. narrow. Therefore, they can also "fail" if the gripper comes to a halt too early, that is $\delta > \Delta_{min}$. This will result in a reinforcement R_{ft}.

EGA2 fails if the gripper is fully opened, meaning that no contact was made with the object. This gives a reinforcement R_{fh}.

To detect failure of EGA4, a tactile sensor is required on the side of the "fingers". If, after positioning and opening the gripper, there is no contact between the object and the tactile sensor, the EGA has failed. This results in a reinforcement R_{fc}.

If none of the above situations is encountered, a positive reinforcement R_g is given, and the EGA is considered successful.

A. Computing Action Parameters

Let $\Gamma = \{\Pi_1, \Pi_2\}$ be a primitive pair, $\Lambda(\Pi)$ be the position of Π and $\Theta(\Pi)$ be the orientation of Π, also let Γ_i be the i:th pair. From that we can calculate

$$\begin{aligned}
\mathbf{d} &= \Lambda(\Pi_2) - \Lambda(\Pi_1) \\
\mathbf{n}_1 &= \Theta(\Pi_1) \times \mathbf{d} \\
\mathbf{n}_2 &= \Theta(\Pi_2) \times \mathbf{d} \\
sw &= \begin{cases} -1 & \text{if} \quad \mathbf{n}_1 \cdot \mathbf{n}_2 < 0 \\ 1 & \text{else} \end{cases}
\end{aligned}$$

and with those we calculate the plane $\mathbf{p}$

$$\begin{aligned}
\mathbf{P_p} &= \Lambda(\Pi_1) + frac\mathbf{d}2 \\
\mathbf{n_p} &= \frac{\mathbf{n}_1 + sw\mathbf{n}_2}{\parallel \mathbf{n}_1 + sw\mathbf{n}_2 \parallel}
\end{aligned}$$

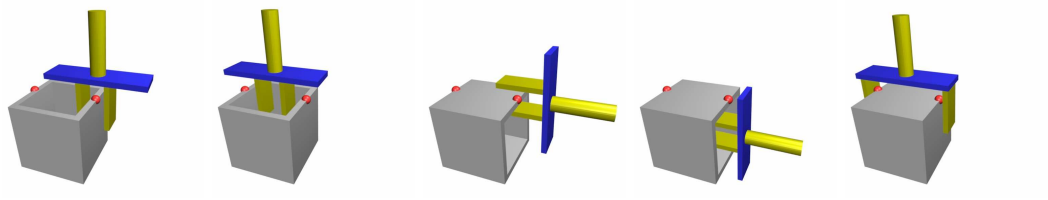
which is used when calculating actions parameters

The parameterization of the EGAs is given with the gripper normal $\mathbf{n}$ and the normal of the surface between the two fingers $\mathbf{a}$ as illustrated in Fig. 6. From this, the yaw, pitch and roll angles can be easily computed.

For EGA1, there will be two possible parameter sets given the primitive pair $\Gamma = \{\Pi_1, \Pi_2\}$. The parameterization is as follows:

$$\begin{aligned}
&\mathbf{p}_{\text{gripper}} = \Lambda(\Pi_i) \\
&\mathbf{n} = \nabla(\mathbf{p}) \\
&\mathbf{a} = \mathbf{perp_n}(\Theta(\Pi_i))/ \parallel \mathbf{perp_n}(\Theta(\Pi_i)) \parallel \quad \text{for } i = 1, 2
\end{aligned}$$

where $\nabla(\mathbf{p})$ is the normal of the plane $\mathbf{p}$ and $\mathbf{perp_u}(\mathbf{a})$ is the projection of $\mathbf{a}$ perpendicular to $\mathbf{u}$. That is $\mathbf{perp_u}(\mathbf{a}) = \mathbf{a} - \mathbf{proj_u}(\mathbf{a})$, where $\mathbf{proj_u}(\mathbf{a})$ is defined according to (1).



(a) EGA1 (b) EGA2 (c) EGA3 (d) EGA4 (e) EGA5

Fig. 5. Elementary grasping actions, EGAs.

For EGA2, there is only one parameter set.

$$\begin{aligned}\mathbf{d} &= \Lambda(\Pi_2) - \Lambda(\Pi_1)\\ \mathbf{p}_{\text{gripper}} &= \Lambda(\Pi_1) + \mathbf{d}/2\\ \mathbf{n} &= \nabla(\mathbf{p})\\ \mathbf{a} &= \mathbf{n} \times \mathbf{d}/ \parallel \mathbf{n} \times \mathbf{d} \parallel\end{aligned}$$

For EGA3, there will be two possible parameter sets for $i = 1, j = 2$ and $i = 2, j = 1$.

$$\begin{aligned}\mathbf{d} &= \Lambda(\Pi_j) - \Lambda(\Pi_i)\\ \mathbf{n} &= \mathbf{d}/ \parallel \mathbf{d} \parallel\\ \mathbf{p}_{\text{gripper}} &= \Lambda(\Pi_i)\\ \mathbf{a} &= \mathbf{n} \times \nabla(\mathbf{p})\end{aligned}$$

For EGA4, there will be two possible parameter sets for $i = 1, j = 2$ and $i = 2, j = 1$. Where ϵ is a step size parameter that will depend on the gripper used.

$$\begin{aligned}\mathbf{d} &= \Lambda(\Pi_j) - \Lambda(\Pi_i)\\ \mathbf{n} &= \mathbf{d}/ \parallel \mathbf{d} \parallel\\ \mathbf{p}_{\text{gripper}} &= \Lambda(\Pi_i) - \nabla(\mathbf{p}) \cdot \epsilon\\ \mathbf{a} &= \mathbf{n} \times \nabla(\mathbf{p})\end{aligned}$$

EGA5 will have the same parameters as EGA2 except that the gripper will be fully opened.

B. Limiting the Number of Actions

For a typical scene, the number of coplanar pairs of primitives is in the order of $10^3 - 10^4$. Given that each coplanar relationship gives rise to 8 different grasps from the five different categories, it is obvious that the number of suggested actions must be further constrained. Another problem is that coplanar structures occur frequently in natural scenes and only a small set of them suggest feasible actions, e.g. objects placed on a table create a lot of 3D line structures coplanar to the table but can not grasped directly by a grasping direction normal to the table. In addition, there exist many coplanar pairs of primitives affording similar grasps.

To overcome some of the above problems, we make use of the structural richness of the primitives. First, their embedding into collinear groups naturally clusters the grasping hypotheses into sets of redundant grasps from which only one needs to be tested. Furthermore, co–colority, gives an additional hypothesis for a potential grasp.

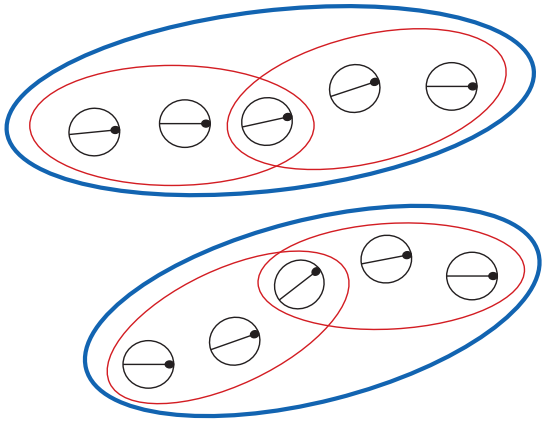


Fig. 7. Small overlapping groups form large groups

1) Using Grouping Information: From the 2D primitives (before stereo reconstruction) collinear neighbors can be found. The collinear neighbors can be mapped to corresponding 3D primitives. These small neighborhoods form the set of *small groups*, $\{g_1, g_2, ..., g_N\}$. The *large groups*, $\{G_1, G_2, ..., G_M\}$, are formed by the grouping of the small groups overlapping each other, Fig. 7 such that if Π_i and Π_j are part of group g_x and Π_j and Π_k is part of group g_y then g_y and g_x is part of the same large group G_z. The result is that the large groups are separated meaning that a primitive that exist in group G_X can not exist in any other group G_Y. Using this grouping information it is possible to add additional constraints on the generation of EGA s.

First, all primitives that are not part of a sufficiently large group G_i are discarded. Secondly, the relations co–planarity and co–colority between small groups of primitives are computed such that primitive $\Pi_i \in g_x$ and $\Pi_j \in g_y$ are only considered to have a co–planarity or co–colority relation if all primitives in g_x are coplanar or cocolor w.r.t all primitives in g_y. Finally, it is possible to constrain the generation of EGAs to only one EGA of each type for each large group.

VI. Experimental Evaluation

Fig. 9, Fig. 10 and Fig. 11 show some of the grasps generated for the scenes evaluated here. Fig. 8 shows visual features generated by the stereo system and a selection of generated actions. Fig. 9 shows a simple plate structure for which the outer contour is generated since the object is

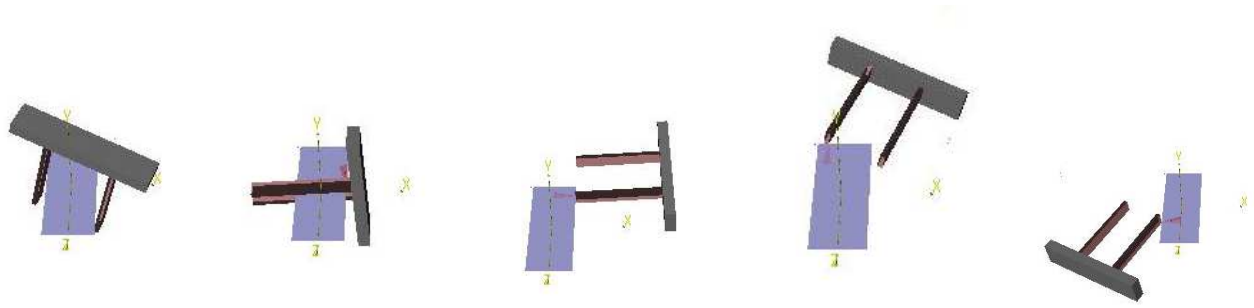


Fig. 9. Examples of tested grasps on a plate (from left): successful grasp using EGA5, and a few early failures using EGA1, EGA3 and EGA5, res5respectively.

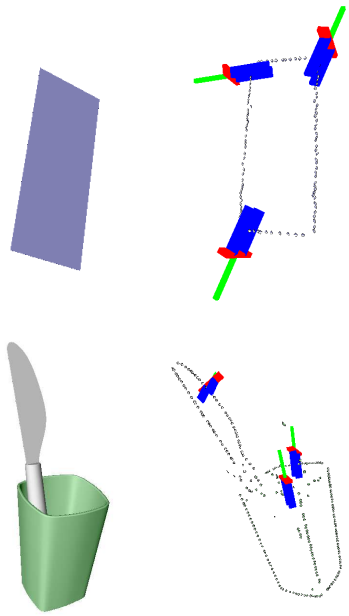


Fig. 8. Two example scenes designed for testing and a selection of the generated actions.

Scene	gr	pl+gr	col+gr	gr+pl+col
Plane	70% (7/10)	83% (5/6)	57% (4/7)	100% (5/5)
Cup	26% (17/66)	38% (14/37)	27% (13/49)	33% (8/24)
Cup/Kn	31% (14/45)	28% (9/32)	31% (11/35)	25% (5/20)
3 objects	8% (33/434)	9% (9/98)	13% (18/139)	15% (8/53)

TABLE I

EXPERIMENTAL EVALUATION OF THE GRASP SUCCESS RATE WHERE THE FOLLOWING NOTATION IS USED: PL (CO-PLANARITY), GR (GROUPING), CL (CO-COLORITY) AND (SUCCESFULL/TESTED) GRASPS.

homogeneous in texture. Fig. 10 shows a scene with a single, but a more complex object than the previous one. Fig. 11 shows two scenes with two (cup and knife) and three objects (box, cup and bottle9.

On each of the scene, after the spatial primitives have been extracted, elementary actions shown in Fig. 5 are tested. There are few reasons for which a certain grasp may fail:

- The system does not have the knowledge of whether the object is hollow or not, so testing EGA2 will results with a collision and thus failure.
- Since no surface is reconstructed, EGA1 will fail for hollow objects which are grasped from “below”.
- If the hand, during the approach, detects a collision on one of the fingers, the grasping process is stopped. In reality, this grasp may happen to be successful anyway if the object is moved so that it is centered between the fingers.

Table I summarizes the results for the generated success rate regarding a number of successful grasps given no knowledge of the object shape. We note that the results are a summary of an extensive experimental evaluation since, given different types and combinations of spatial primitives all generated actions had to be evaluated. It can be seen that for a scene of low complexity (plate) the average number of successful grasps is close to 80%. For more complex scenes this number is dependant on the number and type of objects. It is also important to note not only the percentage but the number of evaluated grasps. Although, in some cases, the success rate is lower when primitives are integrated, there are much fewer hypotheses tested. These results should also be considered together with the results presented in Table II where we show how the integration of grouping, co-colority and co-planarity affects the number of generated hypotheses (affordances). Another thing to point out related to Table I is that most of the unsuccessful grasps happened due to an “early failure” such as that a contact was detected before the grasp was executed. Again, this failure may in some cases result with a successful grasp anyway. Another big source of failure was that there was nothing to lift, i.e. EGA3 could not have been applied.

VII. CONCLUSIONS

Robots should be able to extract more knowledge through their interaction with the environment. The basis for this interaction should not be a detailed model of the environment and lots of *a-priori* knowledge but the robot should be engaged in an exploration process through which it can generate more knowledge and more complex representations. In this paper, we have presented one of the building blocks necessary in such a system.

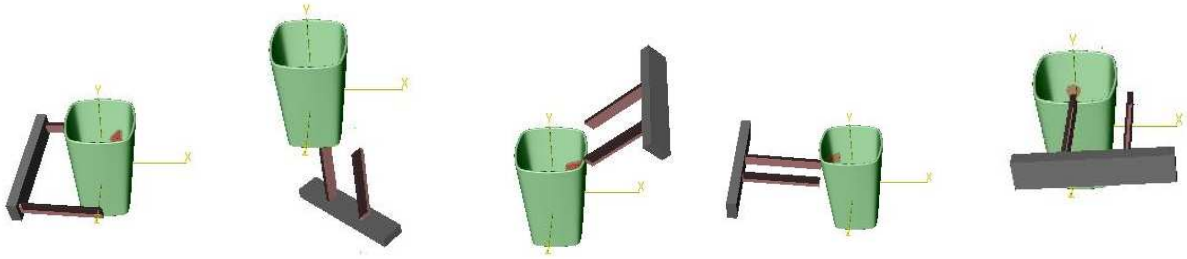


Fig. 10. Examples of tested grasps on a cup (from left): a successful grasp using EGA1, and a few early failures using EGA1, EGA1, EGA2 and EGA3, respectively.

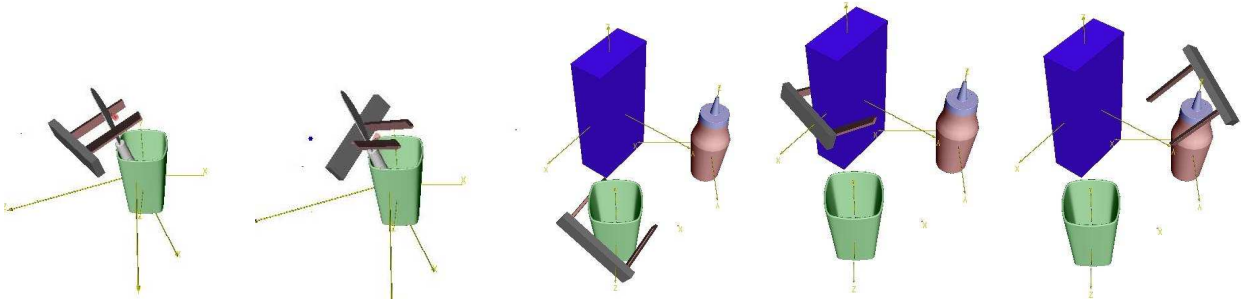


Fig. 11. Examples of successful grasps with two and three objects.

Scene	(no gr)	(no gr)+pl	(no gr)+col	(no gr)+pl+col
Plane	46 224	35 608	38 512	30 224
Cup	172 224	96 112	89 392	56 120
Cup/knife	269 360	140 920	139 136	79 104
3 objects	927 368	303 960	315 336	166 008

Scene	gr	gr+pl	gr+col	gr+pl+col
Plane	80	48	56	40
Cup	528	296	392	192
Cup/knife	360	256	280	160
3 objects	3472	784	1112	424

TABLE II

THE NUMBER OF GENERATED ACTION HYPOTHESES WHERE THE FOLLOWING NOTATION IS USED: NO GR (NO GROUPING), PL (CO-PLANARITY), GR (GROUPING), CL (CO-COLORITY).

In particular, we have designed an early grasping system, based on a set of innate reflexes and knowledge about its embodiment. We relied on 3D information based on primitives extracted online and showed how the structural richness of primitives can be used for an efficient reduction of grasping hypotheses while keeping relevant ones. Rather than dealing with high quality grasps on a constrained set of known objects, we have demonstrated that the system is able of generating a certain percentage of successful grasps on arbitrary objects. This is important for our future research that will develop complex learning schemes aiming at more sophisticated grasping strategies and knowledge representation.

ACKNOWLEDGMENT

This work has been supported by EU through the project PACO-PLUS, FP6-2004-IST-4-27657.

REFERENCES

[1] A. Stoytchev, "Behavior-Grounded Representation of Tool Affordances," in *IEEE International Conference on Robotics and Automation*, pp. 3060–3065, 2005.

[2] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, "Learning About Objects Through Action - Initial Steps Towards Artificial Cognition," in *IEEE International Conference on Robotics and Automation*, pp. 3140–3145, 2003.

[3] P. Azad, T. Asfour, and R. Dillmann, "Combining appearance-based and model-based methods for real-time object recognition and 6d localization," in *IEEE International Conference on Intelligent Robots and Systems*, 2006.

[4] N. Krüger, M. Lappe, and F. Wörgötter, "Biologically motivated multi-modal processing of visual primitives," *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, vol. 1, no. 5, pp. 417–428, 2004.

[5] N. Pugeault, F. Wörgötter, and N. Krüger, "Multi-modal scene reconstruction using perceptual grouping constraints," in *Proceedings of the 5th IEEE Computer Society Workshop on Perceptual Organization in Computer Vision, (in conjunction with IEEE CVPR 2006)*, 2006.

[6] A. T. Miller and P. Allen, "Graspit!: A versatile simulator for grasping analysis," in *ASME International Mechanical Engineering Congress and Exposition*, 2000.

[7] I. Kamon, T. Flash, and S. Edelman, "Learning Visually Guided Grasping: A Test Case in Sensorimotor Learning," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 28, no. 3, pp. 266–276, 1998.

[8] B. Rössler, J. Zhang, and A. Knoll, "Visual Guided Grasping of Aggregates using Self-Valuing Learning," in *IEEE International Conference on Robotics and Automation*, pp. 3912– 3917, 2002.

[9] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *IEEE International Conference on Robotics and Automation*, pp. 348–353, 2000.

[10] D. Ding, Y.-H. Liu, and S. Wang, "Computing 3-d optimal formclosure grasps," in *IEEE International Conference on Robotics and Automation*, pp. 3573 – 3578, 2000.

[11] A. Hauck, J. Rüttinger, M. Sorg, and G. Färber, "Visual Determination of 3D Grasping Points on Unknown Objects with a Binocular Camera System," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 272–278, 1999.

[12] M. Rutishauser and M. Stricker, "Searching for Grasping Opportunities on Unmodeled 3D Objects," in *British Machine Vision Conference*, pp. 277 – 286, 1995.

[13] A. Morales, G. Recatalá, P. J. Sanz, and Á. P. del Pobil, "Heuristic Vision-Based Computation of Planar Antipodal Grasps on Unknown Objects," in *IEEE International Conference on Robotics and Automation*, pp. 583– 588, 2001.

[14] A. T. Miller, S. Knoop, and H. I. C. P.K. Allen, "Automatic grasp planning using shape primitives," in *IEEE International Conference on Robotics and Automation*, pp. 1824–1829, 2003.

[15] N. S. Pollard, "Closure and quality equivalence for efficient synthesis of grasps from examples," *International Journal of Robotic Research*, vol. 23, no. 6, pp. 595–613, 2004.

[16] A. Morales, E. Chinellato, A. H. Fagg, and A. del Pobil, "Using experience for assessing grasp reliability," *International Journal of Humanoid Robotics*, vol. 1, no. 4, pp. 671–691, 2004.

[17] R. Platt Jr, A. H. Fagg, and R. A. Grupen, "Extending fingertip grasping to whole body grasping," in *International Conference on Robotics and Automation*, pp. 2677 – 2682, 2003.

[18] N. S. Pollard, "Parallel methods for synthesizing whole-hand grasps from generalized prototypes," *PhD thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology*, 1994.

[19] http://www.barrett.com/robot/products/hand/handfram.htm.

[20] N. Krüger and F. Wörgötter, "Multi-modal primitives as functional models of hyper-columns and their use for contextual integration," *International Symposium on Brain, Vision and Artificial Intelligence, Lecture Notes in Computer Science, Springer, LNCS 3704*, pp. 157–166, 2005.

[21] N. Krüger and M. Felsberg, "An explicit and compact coding of geometric and structural information applied to stereo matching," *Pattern Recognition Letters*, vol. 25, no. 8, pp. 849–863, 2004.

Robotics Group
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Technical Report no. 2007 – 3

Perceptual Operations and Relations between 2D or 3D Visual Entities

Sinan Kalkan, Nicolas Pugeault, Mogens Christiansen, Norbert Krüger

January 22, 2007

Title	Perceptual Operations and Relations between 2D or 3D Visual Entities
Author(s)	Sinan Kalkan, Nicolas Pugeault, Mogens Christiansen, Norbert Krüger
Publication History	

Abstract

In this paper, we present a set of perceptual relations, namely, co-colority, co-planarity, collinearity and symmetry that are defined between multi-modal visual features that we call *primitives*.

1 Introduction

According to Marr's paradigm [29], vision involves extraction of meaningful representations from input images, starting at the pixel level and building up its interpretation more or less in the following order: local filters, extraction of important features, the $2\frac{1}{2}$-D sketch and the 3-D sketch.
There is psychophysical evidence and evidence from the statistical properties of natural images that the human visual system utilizes a set of visual-entity-combining processes, called *perceptual organization* in the literature, for forming bigger, sparser and more complete interpretations of the scene (see, *e.g.*, [18, 19, 35]). Such processes include (i) extraction of the boundary of the objects in the image from the set of unconnected edge pixels or features [3, 8, 10, 21, 27, 31, 39] utilizing Gestalt laws of grouping, and (ii) interpolation and extrapolation of unconnected sparse 3D entities for forming more complete 3D surfaces (see, *e.g.*, [13]) utilizing the relations between the 3D entities. Gestalt principles include collinearity, proximity, common fate and similarity whereas inference of 3D surfaces from a set of 3D entities include relations like coplanarity, collinearity, co-colority etc. These are essentially second order and higher order relations of local features.
In [26], we have introduced a specific form of a local descriptor that we call a 'multi–modal primitive' (see section 2) and which can be seen as a functional abstraction of a hypercolumn (see [24]). We distinguish between 2D primitives describing local image information and 3D primitives covering local 3D scene information in a condensed symbolic way.
These primitives serve as a basis for an early cognitive vision system [23, 26, 33] in which operations and relations on these primitives realizing perceptual grouping principles are used in different contexts (see [26] for applications). We have utilized these relations for different problems including stereo [34], RBM [32], estimation of initial grasping reflexes from stereo [5], estimation of depth at homogeneous image structures [16], and analysis of second-order relations between 3D features [17].
In this paper, we present the set of 2D and 3D relations defined upon the primitives. These relations include collinearity, cocolority, coplanarity and symmetry. Of these relations, collinearity, cocolority and symmetry are defined for 2D as well as 3D primitives whereas by definition, coplanarity is meaningful only for 3D primitives. Table 1 summarizes the relations and on which dimension they are defined.

Relation	2D	3D
co-planarity	$\times$	$\surd$
co-colority	$\surd$	$\surd$
collinearity	$\surd$	$\surd$
symmetry	$\surd$	$\surd$

Table 1: The relations and in which dimension they are defined.

This paper does not focus on any specific application domain but provides a technically detailed definition of these relations that are usually not described in such detail in publications making use of them.
The paper is organized as follows: In section 2, we briefly introduce our visual features, namely primitives. In section 3, we describe our definitions of perceptual relations between the visual primitives. In section 5, we conclude the paper.

2 Primitives

Numerous feature detectors exist in the literature (see [30] for a review). Each feature based approach can be divided into an interest point detector (e.g. [14, 4]) and a descriptor describing a local patch of the image at this location, that can be based on histograms (e.g. [6, 30]), spatial frequency [20], local derivatives [15, 11, 1] steerable filters [12], or invariant moments ([28]). In [30] these different descriptors have been compared, showing a best performance for SIFT-like descriptors.
The primitives we will be using in this work are local, multi–modal edge descriptors that were introduced in [25]. In contrast to the above mentioned features these primitives focus on giving a semantically and geometrically meaningful description of the local image patch. The importance of such a semantic grounding of features for a general purpose vision front–end, and the relevance of edge–like structures for this purposes were discussed in [9].
The primitives are extracted sparsely at locations in the image that are the most likely to contain edges. This likelihood is computed using the intrinsic dimensionality measure proposed in [22]. The sparseness is assured using a classical winner take all operation, insuring that the generative patches of the primitives do not overlap (for details, see [26]). Each of the primitive encodes the image information contained by a local image patch. Multi–modal information is gathered from this image patch, including the position $\boldsymbol{m}$ of the centre of the patch, the orientation θ of the edge, the phase ω of the signal at this point, the colour $\boldsymbol{c}$ sampled over the image patch on both sides of the edge and the local optical flow $\boldsymbol{f}$. Consequently a local image patch is described by the following multi–modal vector:

$$\boldsymbol{\pi} = (\boldsymbol{m}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T, \tag{1}$$

that we will name *2D primitive* in the following.
Note that these primitives are of lower dimensionality than, e.g., SIFT (10 vs. 128) and therefore suffer of a lesser distinctiveness. Nonetheless, as shown in [34] that they are distinctive enough for a reliable stereo matching if the epipolar geometry of the cameras is known. Furthermore, their semantic in terms of geometric and appearance based information allow for a good description of the scene content. It has been previously argued in [9] that edge pixels contain all important information in an image. As a consequence, the ensemble of all primitives extracted from an image describe the shapes present in this image.
Advantageously, the rich information carried by the 2D–primitives can be reconstructed in 3D, providing a more complete scene representation. Having geometrical meaning for the primitive allows to describe the relation between proximate primitives in terms of perceptual grouping.
In a stereo scenario 3D primitives can be computed from the correspondences of 2D primitives (see figure 1 and [34]):

$$\boldsymbol{\Pi} = (\boldsymbol{M}, \Theta, \boldsymbol{\Omega}, \boldsymbol{C})^T, \tag{2}$$

such that we have a projection relation:

$$\mathcal{P} : \boldsymbol{\Pi} \rightarrow \boldsymbol{\pi}\,. \tag{3}$$

3 Relations

In this section, we present collinearity, cocolority, coplanarity and symmetry relations that are defined on our visual features.

3.1 Collinearity in 2D and 3D

As the primitives are local contour descriptors, scene contours are expected to be represented by strings of primitives that are locally close to collinear. In the following, we will explain methods for grouping 2D and 3D primitives into contours.

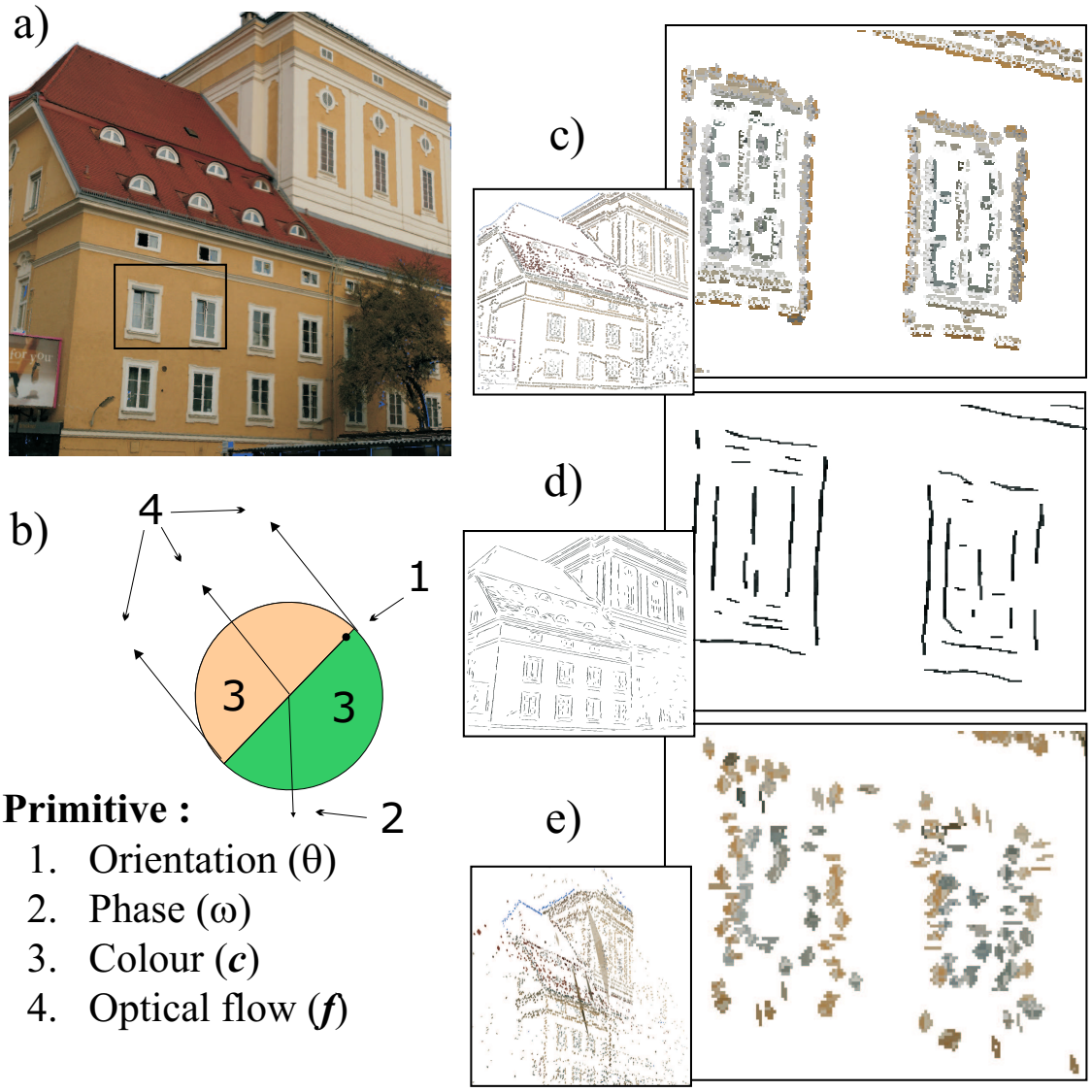


Figure 1: Illustration of the primitive extraction process from a video sequence. The 2D–primitives extracted from the input image (a) (see section 2), and finally the 3D–primitives reconstructed from the stereo–matches as described as described in [34]. **(a)** An example input image. **(b)** A graphic description of the 2D–primitives. **(c)** A magnification of the image representation. **(d)** Perceptual grouping of the primitives as described in [34]. **(e)** The reconstructed 3D entities. Note that the structure reconstructed is quite far from the cameras, leading to a certain imprecision in the reconstruction of the 3D–primitives. A simple scheme addressing this problem is described in [34].

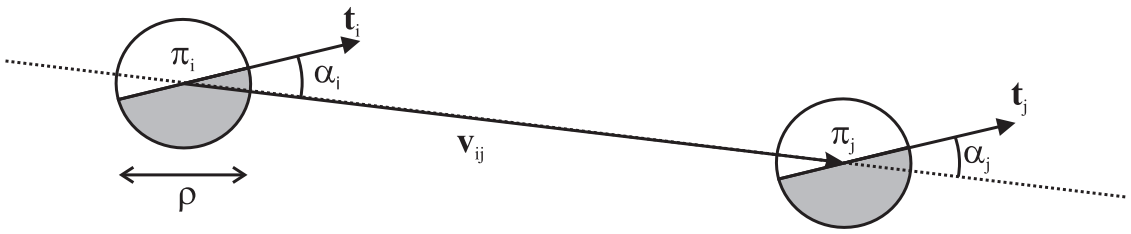


Figure 2: Illustration of the values used for the collinearity computation. If we consider two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$, then the vector between the centres of these two primitives is written $\boldsymbol{v}_{ij}$, and the orientations of the two primitives are designated by the vectors t_i and t_j, respectively. The angle formed by $\boldsymbol{v}_{ij}$ and t_i is written α_i, and between $\boldsymbol{v}_{ij}$ and t_j is written α_j. ρ is the radius of the image patch used to generate the primitive.

3.1.1 Collinearity in 2D

In the following, $c(l_{i,j})$ refers to the likelihood for two primitives π_i and π_j to be *linked*: i.e. grouped to describe the same contour.
Position and orientation of primitives are intrinsically related. As primitives represent local edge estimators, their positions are points along the edge, and their orientation can be seen as a tangent at such a point. The estimated likelihood of the contour described by those tangents is based upon the assumption that simpler curves are more likely to describe the scene structures, and highly jagged contours are more likely to be manifestations of erroneous and noisy data.
Therefore, for a pair of primitives π_i and π_j in image $\boldsymbol{\mathcal{I}}$, we can formulate the likelihood for these primitives to describe the same contour as a combination of three basic constraints on their relative position and orientation — see [34].

Proximity ($c_p[l_{i,j}]$): A contour is more likely if it is described by a dense population of primitives. Large holes in the primitive description of the contour is an indication that there are two contours which are collinear yet different. The proximity constraint is defined by the following equation:

$$c_p[l_{i,j}] = 1 - e^{-\max\left(1-\frac{||\mathbf{v_{i,j}}||}{\rho\tau},0\right)}, \tag{4}$$

where ρ stands for the size of the receptive field of the primitives in pixels; $\rho\tau$ is the size of the neighbourhood considered in pixels; and, $\|\mathbf{v_{i,j}}\|$ is the distance in pixels separating the centres of the two primitives.

Collinearity ($c_{co}[l_{i,j}]$): A contour is more likely to be linear, or to form a shallow curve rather than a sharp one. A sharp curve might be an indication of two intersecting or occluding contours.

$$c_{co}[l_{i,j}] = 1 - \left|\sin\left(\frac{|\alpha_i| + |\alpha_j|}{2}\right)\right|, \tag{5}$$

where α_i and α_j are the angles between the line joining the two primitives centres and the orientation of, respectively, π_i and π_j.

Co–circularity ($c_{ci}[l_{i,j}]$): A contour is more likely to have a continuous, or smoothly changing curvature, rather than a varying one. An unstable curvature is an indicator of a noisy, erroneous or under–sampled contour, all of which are unreliable.

$$c_{ci}[l_{i,j}] = 1 - \left|\sin\left(\frac{\alpha_i + \alpha_j}{2}\right)\right|, \tag{6}$$

Geometric Constraint ($\mathbf{G}_{i,j}$): The combination of those three criteria provided above forms the following *geometric* affinity measure:

$$\mathbf{G}_{i,j} = \sqrt[3]{c_e[l_{i,j}] \cdot c_{co}[l_{i,j}] \cdot c_{ci}[l_{i,j}]}, \tag{7}$$

where $\mathbf{G}_{i,j}$ is the geometric affinity between two primitives π_i and π_j. This affinity represents the likelihood that two primitives π_i and π_i are part of an actual contour of the scene.

Multi–modal Constraint ($\mathbf{M}_{i,j}$): The geometric constraint offers a suitable estimation of the likelihood of the curve described by the pair of primitives. Other modalities of the primitives allow inferring more about the qualities of the physical contour they represent. The colour, phase and optical flow of the primitives further define the properties of the contour, and thus consistency constraints can also be enforced over those modalities. Effectively, the less difference there is between the modalities of two primitives, the more likely that they are expressions of the same contour. In [7], it is already proposed that the intensity can be used as a cue for perceptual grouping; our definition goes beyond this proposal by using a combination of the phase, colour and optical flow modalities of the primitives to decide if they describe the same contour:

$$\mathbf{M}_{i,j} = w_\omega c_\omega[l_{i,j}] + w_{\boldsymbol{c}} c_{\boldsymbol{c}}[l_{i,j}] + w_{\boldsymbol{f}} c_{\boldsymbol{f}}[l_{i,j}], \tag{8}$$

where c_ω is the phase criterion, $c_{\boldsymbol{c}}$ the colour criterion and $c_{\boldsymbol{f}}$ the optical flow criterion. Each of the three w_ω, $w_{\boldsymbol{c}}$ and $w_{\boldsymbol{f}}$ is the relative scaling for each modality, with $w_\omega + w_{\boldsymbol{c}} + w_{\boldsymbol{f}} = 1$.

Primitive Affinity ($\boldsymbol{A}_{i,j}$): The overall affinity between all primitives in an image is formalised as a matrix $\boldsymbol{A}$, where $\boldsymbol{A}_{i,j}$ holds the affinity between the primitives π_i and π_j. We define this affinity from equations 7 and 8, such that (1) two primitives complying poorly with the good continuation rule have an affinity close to zero; and (2) two primitives complying with the good continuation rule yet strongly dissimilar will have only an average affinity. The affinity is formalised as follows:

$$c(l_{i,j}) = \boldsymbol{A}_{i,j} = \sqrt{\mathbf{G}\left(\alpha \mathbf{G}_{i,j} + (1-\alpha)\mathbf{M}_{i,j}\right)}, \tag{9}$$

where α is the weighting of geometric and multi–modal (*i.e.* phase, colour and optical flow) information in the affinity. A setting of $\alpha = 1$ implies that only geometric information (proximity, collinearity and co-circularity) is used, while $\alpha = 0$ means that geometric and multi–modal information are evenly mixed.

3.1.2 Collinearity in 3D

Collinearity in 3D is more difficult to define. Due to the inaccuracy in stereo–reconstruction of 3D position and orientation, it is impossible to apply strong alignment constraints such as the ones we applied in the 2D case. Consequently we will define 3D collinearity as follows:

Definition 1 *Two 3D–primitives Π_i and Π_j are said collinear if the 2D–primitives π_i^x and π_j^x they project onto the camera plane x (defined by a projection relation $\mathcal{P}^x : \Pi_k \rightarrow \pi_k$) are all collinear (according to the definition of 2D–primitive collinearity presented above).*

and therefore in the standard case where we have two stereo cameras labelled l and r we have the following relation:

$$c(L_{i,j}) = c(l^l_{i,j}) \cdot c(l^r_{i,j}). \tag{10}$$

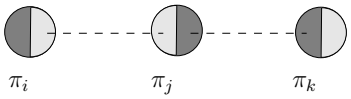


Figure 3: Co–colority of three 2D primitives π_i, π_j and π_k. In this case, π_i and π_j are cocolor, so are π_i and π_k; however, π_j and π_k are not cocolor.

3.2 Cocolority in 2D and 3D

Two spatial primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ are co–color iff their parts that face each other have the same color. In the same way as collinearity, co–colority of two spatial primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ is computed using their 2D projections $\mathcal{P}\Pi_i = \pi_i$ and π_j. We define the co–colority of two 2D primitives π_i and π_j as:

$$coc(\pi_i, \pi_j) = 1 - \mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j),$$

where $\mathbf{c}_i$ and $\mathbf{c}_j$ are the RGB representation of the colors of the parts of the primitives π_i and π_j that face each other; and, $\mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j)$ is Euclidean distance between RGB values of the colors $\mathbf{c}_i$ and $\mathbf{c}_j$. In Fig. 3, a pair of co–color and not co–color primitives are shown.
Euclidean color distance d_c is a simple one compared to color distance metrics developed by different institutes like International Commission on Illumination (CIE). Such metrics are developed to match our perception of colour and are computationally expensive (see, *e.g.*, [38]). For our purposes, Euclidean distance between RGB values is sufficient and can be replaced by a more complicated distance metric, if desired.
3D co-colority is defined as follows:

Definition 2 *Two 3D–primitives Π_i and Π_j are said cocolor if the 2D–primitives π_i^x and π_j^x they project onto the camera plane x (defined by a projection relation $\mathcal{P}^x : \Pi_k \rightarrow \pi_k$) are co-color (according to the definition of 2D–primitive cocolority presented above).*

3.3 Coplanarity

According to [37],

> a set of points in space is coplanar if the points all lie in a geometric plane. For example, three points are always coplanar; but four points in space are usually not coplanar.

Although the definitions are more or less the same, there are different ways to *check* the coplanarity of a set of points [36, 37]. For a set of n points $\mathbf{x}_1 ... \mathbf{x}_n$ where $\mathbf{x}_i = (x_i, y_i, z_i)$, the following methods can be adopted:

- For $n = 4$, $\mathbf{x}_1 ... \mathbf{x}_n$ are coplanar
  - iff the volume of the tetrahedron defined by them is 0 [36], *i.e.*,

$$\begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix} = 0. \tag{11}$$

  - iff the pair of lines determined by the four points are not skew [36]:

$$(\mathbf{x}_3 - \mathbf{x}_1).[(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_4 - \mathbf{x}_3)] = 0. \tag{12}$$

– iff $\mathbf{x}_4$ is on the plane defined by $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$:

$$d(\mathbf{x}_4, P(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)) = 0, \tag{13}$$

where $P(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ is the plane defined by $P(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$, and $d(\mathbf{x}, \mathbf{p})$ is the distance between point $\mathbf{x}$ and plane $\mathbf{p}$.

- For $n > 4$, $\mathbf{x}_1 ... \mathbf{x}_n$ are coplanar iff point-plane distances of $\mathbf{x}_4 .. \mathbf{x}_n$ to the plane defined by $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ are all zero:

$$\sum_{i=4}^{n} d(\mathbf{x}_i, P(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)) = 0. \tag{14}$$

3.3.1 Coplanarity of bounded planes

A bounded plane $\mathbf{p}^b$ is part of the plane $\mathbf{p}$ with a certain size $\mathbf{s}$ and position $\mathbf{x}$. In other words, $\mathbf{p}^b$ is equivalent to $(\mathbf{n}, \mathbf{x}, \mathbf{s})$ where $\mathbf{n}, \mathbf{x}, \mathbf{s}$ are respectively the normal (*i.e.*, orientation), position (*i.e.*, center) and the size of the bounded plane.
As suggested in [17], two bounded planes $\mathbf{p}_1^b, \mathbf{p}_2^b$ are coplanar if:

$$(\alpha(\mathbf{n}_1, \mathbf{n}_2) < T_\alpha) \ \wedge \ (\frac{d(\mathbf{x}_1, \mathbf{p}_2^b)}{d(\mathbf{x}_1,\ \mathbf{x}_2)} < T_d), \tag{15}$$

where $\alpha(\mathbf{n}_1, \mathbf{n}_2)$ is the angle between the two orientations vectors $\mathbf{n}_1$ and $\mathbf{n}_1$, and T_α and T_d are the thresholds.

3.3.2 Coplanarity of 3D primitives

Two spatial primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ are co–planar iff their orientation vectors lie on the same plane, i.e.:

$$cop(\mathbf{\Pi}_i, \mathbf{\Pi}_j) = 1 - |\mathbf{proj}_{t_j \times v_{ij}}(t_i \times v_{ij})|, \tag{16}$$

where $\mathbf{v}_{ij}$ is defined as the vector $(\boldsymbol{M}_i - \boldsymbol{M}_j)$; t_i and t_j denote the vectors defined by the 3D orientations Θ_i and Θ_j, respectively; and $\mathbf{proj}_{\mathbf{u}}(\mathbf{a})$ is defined as:

$$\mathbf{proj}_{\mathbf{u}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{u}}{\| \mathbf{u} \|^2} \mathbf{u}. \tag{17}$$

The co–planarity relation is illustrated in Fig. 4.

3.4 Symmetry in 2D and 3D

Two primitives are symmetric if they are located on two contours which are reflections of each other (see figure 5(a)). This reflective symmetry between two primitives can be measured by utilizing the angles between the orientations of the primitives and the line that joins the centers of the primitives.
Let v_{ij} denote the line joining the centers of the primitives, π_i and π_j, and also ϕ_{ij} and ϕ_{ji} be the angles between v_{ij} and the lines defined by the orientations of π_i and π_j, respectively (see figure 5). Then, two 2D primitives π_i and π_j can be considered symmetric, if $\phi_{ij} = \phi_{ji}$ with a symmetry axis a_{ij} defined as follows:

$$a_{ij} = \begin{cases} L(c_{ij}; \theta_i) & \text{if } \theta_i = \theta_j, \\ L(c_{ij}; \alpha_{ij}), & \text{otherwise,} \end{cases} \tag{18}$$

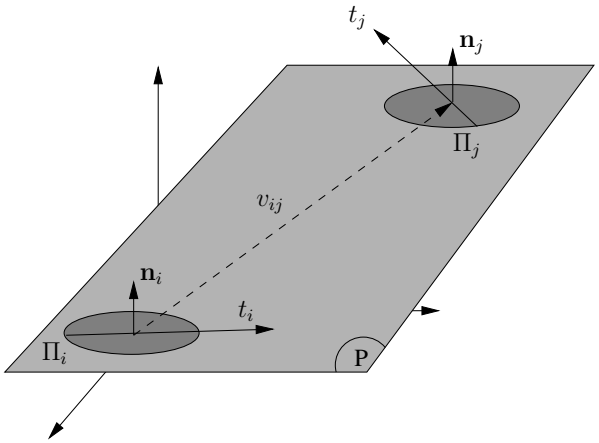


Figure 4: Co–planarity of two 3D primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$. t_i and t_j denote the vectors defined by the 3D orientations Θ_i and Θ_j, respectively.

where $L(x;\theta)$ is a line that goes through a point x with orientation θ; $\mathbf{int}(l_k, l_m)$ is the intersection point of two lines denoted by l_k and l_m; c_{ij} is defined as the mid-point of v_{ij} (*i.e.*, $(\boldsymbol{m}_i + \boldsymbol{m}_j)/2$); and, α_{ij} is the angle of the line that joins the points c_{ij} and $\mathbf{int}(L(\boldsymbol{m}_i;\theta_i), L(\boldsymbol{m}_j;\theta_j))$.
The symmetry axis a_{ij} is undefined if the primitive orientations θ_i and θ_j, and v_{ij} are all parallel, which is the case when both primitives are located on the same linear segment of a contour. This is the case for π_j and π_k in figure 5(b) and 5(c). If the symmetry axis a_{ij} is undefined, a primitive pair should not be regarded as symmetric, but collinear.
Figure 5 illustrates a few symmetric and non-symmetric primitives. In figure 5(b) and 5(c), as the primitives π_j and π_k are on the same contour, a_{ij} is parallel with the primitive orientations θ_j, θ_k and v_{jk}.
Taking collinearity into account, symmetry between two primitives π_i and π_j is defined as follows:

$$sym(\pi_i, \pi_j) = \begin{cases} 0 & \text{if } c_{co}[l_{i,j}] > T_c, \\ 1 - |\sin(\phi_{ij} - \phi_{ji})| & \text{otherwise,} \end{cases} \tag{19}$$

where $c_{co}[l_{i,j}]$ is the collinearity relation and T_c is a threshold, determining if π_i and π_j are collinear.
Like collinearity and co–colority, the symmetry of two 3D primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ is computed using their 2D projections π_i and π_j:

Definition 3 *Two 3D–primitives $\mathbf{\Pi}_i$ and $\mathbf{\Pi}_j$ are said to be symmetric if the 2D–primitives π_i^x and π_j^x they project onto the camera plane x (defined by a projection relation $\mathcal{P}^x : \Pi_k \rightarrow \pi_k$) are symmetric (according to the definition of 2D–primitive symmetry presented above).*

4 Results

In figure 6, the coplanarity, cocolority and collinearity relations are shown for two different example scenes shown in figure 6(a) and (b). The results are from our 3D display tool called *Wanderer*, and for computational reasons, 3D primitives are shown in squares. The relations are displayed only for a primitive which is selected with the mouse as showing relations between all primitives disables visibility.
From the figure we see that coplanarity is a more common relation than cocolority or collinearity. This suggests that coplanarity alone is not directly usable for analysis or applications in 3D, and it needs to be accompanied with other relations as proposed and utilized in [2, 16].

5 Conclusion

In this paper, we presented cocolority, coplanarity, collinearity and symmetry relations defined on multi-modal visual features, called primitives.
Such relations have been utilized in different perceptual organization problems as well as analysis of how the natural scenes are structured (see, *e.g.*, ([3, 8, 10, 13, 16, 17, 21, 27, 31, 34, 39]), and the importance of such relations, as well as their psychophysical and biological plausibility have been acknowledged in the literature (see, *e.g.*, [18, 19, 35]).

6 Acknowledgments

We would like to thank Florentin Wörgötter and Daniel Aarno for their fruitful contributions. This work is supported by the Drivsco and the PACO+ projects.

References

[1] A. Baumberg. Reliable Feature Matching across Widely Separated Views. In *Proc. Conf. Computer Vision and Patter Recognition*, pages 774–781, 2000.

[2] D. Aarno, J. Sommerfeld, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger. Model-independent grasping initializing object-model learning in a cognitive architecture. *IEEE International Conference on Robotics and Automation (ICRA), Workshop: From features to actions - Unifying perspectives in computational and robot vision*, 2007.

[3] E. Brunswik and J. Kamiya. Ecological cue–validity of 'proximity' and of other Gestalt factors. *American Journal of Psychologie*, LXVI:20–32, 1953.

[4] Cordelia Schmid and Roger Mohr and Christian Baukhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.

[5] J. S. D. Aarno, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger. Early reactive grasping with second order 3d feature relations. *IEEE Conference on Robotics and Automation (submitted)*, 2007.

[6] David G. Lowe. Distinctive Image Features from Scale–Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004.

[7] J. Elder and R. Goldberg. Inferential reliability of contour grouping cues in natural images. *Perception Supplement*, 27, 1998.

[8] J. Elder and R. Goldberg. Ecological statistics of Gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2(4):324–353, 8 2002.

[9] J. H. Elder. Are edges incomplete ? *International Journal of Computer Vision*, 34:97–122, 1999.

[10] J. H. Elder, A. Krupnik, and L. A. Johnston. Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(25):1–14, 2003.

[11] Frederik Schaffalitzky and Andrew Zisserman. Multi–view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". *Lecture Notes in Computer Science*, 2350:414–431, 2002. in Proceedings of the BMVC02.

[12] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE-PAMI*, 13(9):891–906, 1991.

[13] W. E. L. Grimson. A Computational Theory of Visual Surface Interpolation. *Royal Society of London Philosophical Transactions Series B*, 298:395–427, Sept. 1982.

[14] C. G. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[15] J. J. Koenderink and A. J. van Doorn. Representation of Local Geometry in the Visual System. *Biological Cybernetics*, 55:367–375, 1987.

[16] S. Kalkan, F. Wörgötter, and N. Krüger. Depth prediction at homogeneous image structures. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-2, 2007.

[17] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of second-order relations of 3d structures. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[18] K. Koffka. *Principles of Gestalt Psychology*. Lund Humphries, London, 1935.

[19] K. Köhler. *Gestalt Psychology: An introduction to new concepts in psychology*. New York: Liveright, 1947.

[20] P. Kovesi. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3):1–26, 1999.

[21] N. Krüger. Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129, 1998.

[22] N. Krüger and M. Felsberg. A continuous formulation of intrinsic dimension. *Proceedings of the British Machine Vision Conference*, pages 261–270, 2003.

[23] N. Krüger, M. V. Hulle, and F. Wörgötter. Ecovision: Challenges in early-cognitive vision. *International Journal of Computer Vision*, accepted.

[24] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 1(5):417–428, 2004.

[25] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence & the Simulation of Behavious, AISB Journal*, 1(5):417–427, 2004.

[26] N. Krüger, N. Pugeault, and F. Wörgötter. Multi-modal primitives: Local, condensed, and semantically rich visual descriptors and the formalization of contextual information. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-4, 2007.

[27] N. Krüger and F. Wörgötter. Statistical and deterministic regularities: Utilisation of motion and grouping in biological and artificial visual systems. *Advances in Imaging and Electron Physics*, 131:82–147, 2004.

[28] Luc Van Gool and Theo Moons and Dorin Ungureanu. Affine / Photometric Invariants for Planar Intensity Patterns. *Lecture Notes In Computer Science*, 1064:642–651, 1996. in Proceedings of the 4th European Conference on Computer Vision — Volume 1.

[29] D. Marr. *Vision: A computational investigation into the human representation and processing of visual information*. Freeman, 1977.

[30] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[31] N. Pugeault, N. Krüger, and F. Wörgötter. A non-local stereo similarity based on collinear groups. *Proceedings of the Fourth International ICSC Symposium on Engineering of Intelligent Systems*, 2004.

[32] N. Pugeault, N. Krüger, and F. Wörgötter. Rigid body motion estimation in an early cognitive vision framework. In *IEEE Advances In Cybernetic Systems*, 2006.

[33] N. Pugeault, F. Wörgötter, , and N. Krüger. Disambiguation .....

[34] N. Pugeault, F. Wörgötter, and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proc. IEEE Workshop on Perceptual Organization in Computer Vision (in conjunction with CVPR'06)*, 2006.

[35] S. Sarkar and K. Boyer. *Computing Perceptual Organization in Computer Vision*. World Scientific, 1994.

[36] E. W. Weisstein. Coplanar. from mathworld–a wolfram web resource, 2006. `http://mathworld.wolfram.com/Coplanar.html`.

[37] Wikipedia. Coplanarity — wikipedia, the free encyclopedia, 2006. `http://en.wikipedia.org/w/index.php?title=Coplanarity&oldid=37490165`.

[38] X. Zhang and B. A. Wandell. Color image fidelity metrics evaluated using image distortion maps. *Signal Processing*, 70(3):201–214, 1998.

[39] S. C. Zhu. Embedding gestalt laws in markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1170–1187, 1999.

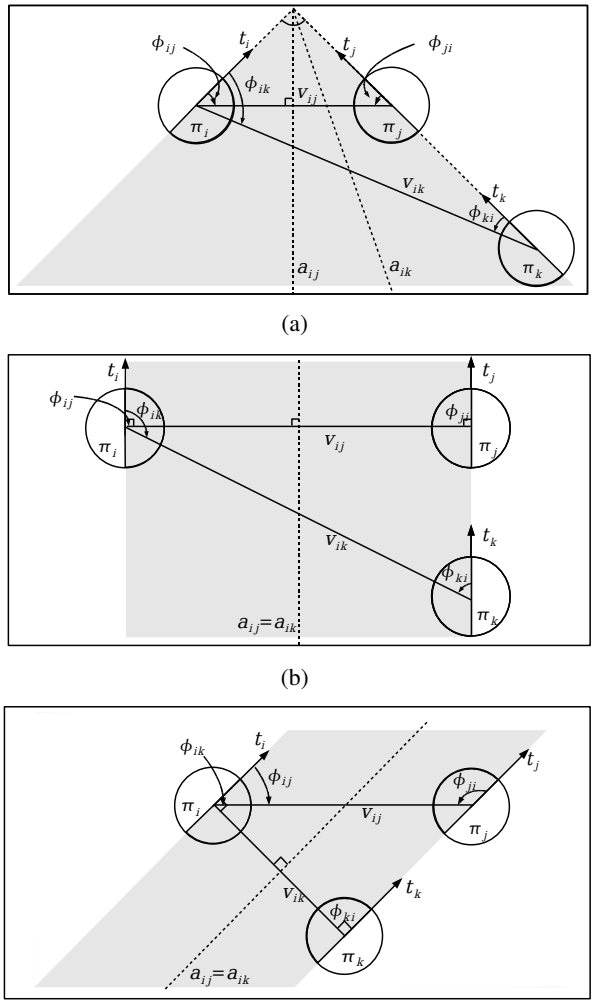


Figure 5: Illustration of the definition of symmetry. t_i, t_j and t_k denote the vectors defined by the orientations θ_i, θ_j and θ_k, respectively. Primitives π_i and π_j are symmetric in (a) and (b), but not in (c). π_i and π_k are symmetric in (c), but not in (a) or (b).

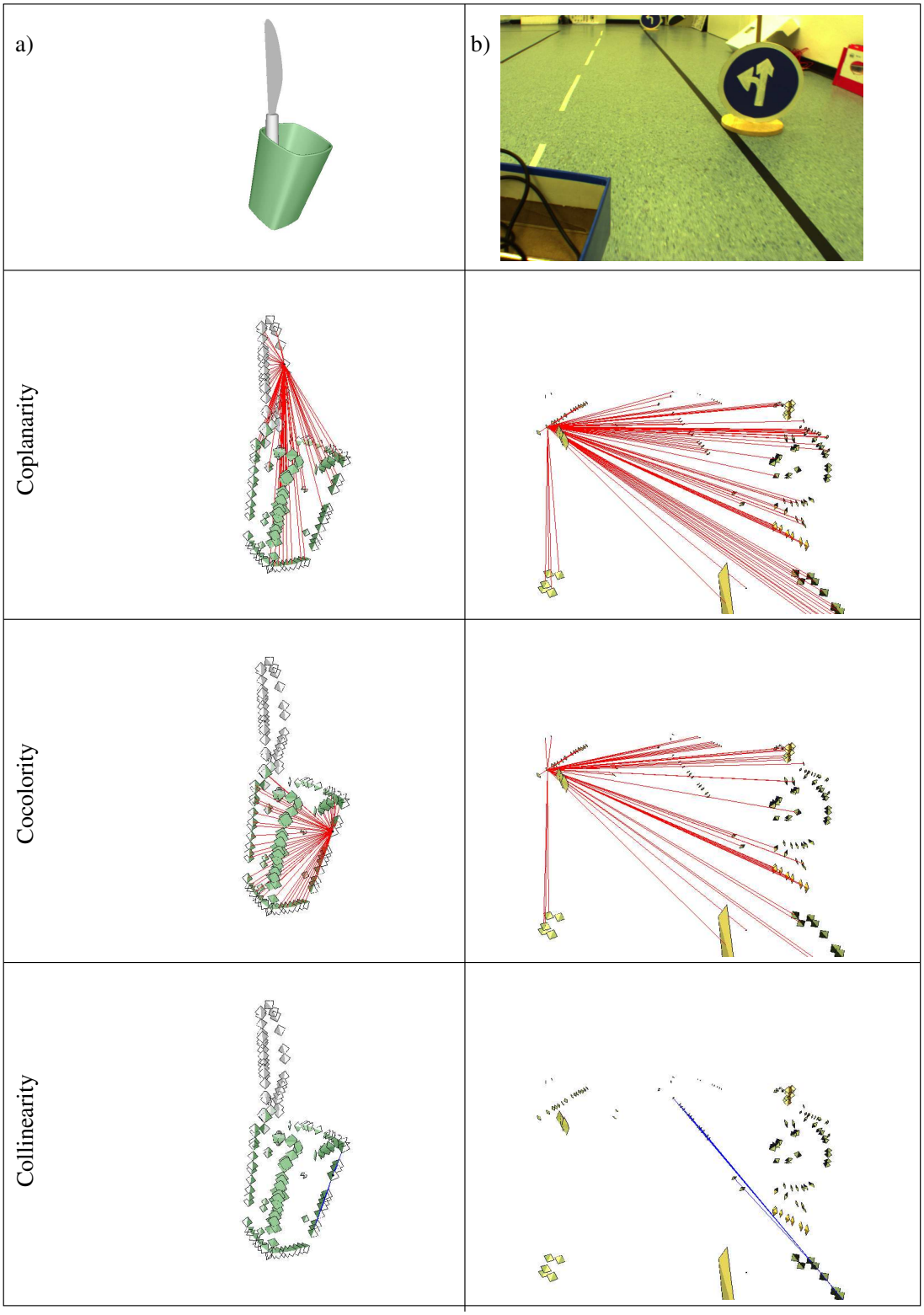


Figure 6: The coplanarity, cocolority and collinearity relations on two different examples shown in (a) and (b). The results are from our 3D display tool called *Wanderer*, and for the sake of speed, 3D primitives are shown in squares. The relations are shown only for a selected primitive as showing relations between all primitives disables visibility.

Published on IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2006.

Statistical Analysis of Local 3D Structure in 2D Images

Sinan KALKAN
Bernstein Centre for Computational Neuroscience,
University of Göttingen, Germany
sinan@chaos.gwdg.de

Florentin Wörgötter
Bernstein Centre for Computational Neuroscience,
University of Göttingen, Germany
worgott@chaos.gwdg.de

Norbert Krüger
Cognitive Vision Group,
Aalborg University Copenhagen, Denmark
nk@media.aau.dk

Abstract

*For the analysis of images, a deeper understanding of their intrinsic structure is required. This has been obtained for 2D images by means of statistical analysis [15, 18]. Here, we analyze the relation between local image structures (*i.e.*, homogeneous, edge-like, corner-like or texture-like structures) and the underlying local 3D structure, represented in terms of continuous surfaces and different kinds of 3D discontinuities, using 3D range data with the true color information. We find that homogeneous image patches correspond to continuous surfaces, and discontinuities are mainly formed by edge-like or corner-like structures. The results are discussed with regard to existing and potential computer vision applications and the assumptions made by these applications.*

1. Introduction

With the notion that the human visual system is adapted to the statistics of the environment [2, 13, 15, 18, 22, 21] and its successful applications to grouping, object recognition and stereo [3, 4, 20, 29] the analysis, and the usage of natural image statistics has become an important focus of vision research. Moreover, with the advances in technology, it has been also possible to analyze the underlying 3D world using 3D range scanners [10, 11, 19, 27].

In this paper, we analyze the relation between local image structures (*i.e.*, homogeneous, edge-like, corner-like or texture-like structures) and the underlying local 3D structure using 3D range data with the true color information.

There have been only a few studies that have analyzed the 3D world from range data [10, 11, 19, 27]. In [27], the distribution of roughness, size, distance, 3D orientation, curvature and independent components of surfaces was analyzed. Their major conclusions were: (1) local 3D patches tend to be saddle-like, and (2) natural scene geometry is quite regular and less complex than luminance images. In [11], the distribution of 3D points was analyzed using co-occurrence statistics and 2D and 3D joint distributions of Haar filter reactions. They showed that range images are much simpler to analyze than optical images and that a 3D scene is composed of piecewise smooth regions. In [19], the correlation between light intensities of the image data and the corresponding range data as well as surface convexity were investigated. They could justify the event that brighter objects are closer to the viewer, which is used by shape from shading algorithms in estimating depth. In [9, 10], range image statistics were analyzed for explanation of several visual illusions.

Our analysis differs from these works. For 2D local image patches, existing studies have only considered light intensity. As for 3D local patches, the most complex considered representation have been the curvature of the local 3D patch. In this work, however, we create a higher-order representation of the 2D local image patches and the 3D local patches; we measure 2D local image patches using homogeneous, edge-like, corner-like or texture-like structures, and 3D local patches using continuous surfaces and different kinds of 3D discontinuities. By this, we relate established local image structures to their underlying 3D structures.

By creating 2D and 3D representations of the local structure, we compute the conditional probability $P(\text{3D Structure} \mid \text{2D Structure})$. Using this probability, we quantify some assumptions made by the studies that reconstruct the 3D world from dense range data. For example, we could show that the depth distribution varies significantly for different visual features, and we could quantify already established inter-dependencies such as 'no new is

good news' [6]. This work also supports the understanding of how intrinsic properties 2D–3D relations can be used for the reconstruction of depth, for example, by using statistical priors in the formalisation of depth cues.

The paper is organized as follows: In section 2, we define the types of local image structures and local 3D structures that we extract for our analysis. In section 3, we introduce a continuous classifier for local 2D structures. In section 4, we outline our methods for measuring the 3D structure of a 3D point. We present and discuss our results in section 5. Finally, we conclude the paper in section 6.

2. Local 2D and 3D Structures

We distinguish between the following local 2D structures:

- Homogeneous image patches: Homogeneous patches are signals of uniform intensities.
- Edge–like structures: Edges are low-level structures which constitute the boundaries between homogeneous or texture-like signals (see, *e.g.*, [14, 17] for their importance in vision).
- Corners: Corners are signals where two or more edge-like structures with significantly different orientations intersect (see, *e.g.*, [7, 23, 24] for their importance in vision).
- Texture: Although there is not a widely-agreed definition, textures are often defined as signals which consist of repetitive, random or directional structures (for their analysis, extraction and importance in vision, see *e.g.*, [26]).

Locally, it is hard to distinguish between these structures, and there are structures that carry mixed properties of the 'ideal' cases. The classification of the features outlined above is discrete. However, a discrete classification may cause problems as the inherent properties of "mixed" structures are lost in the discretization process. Instead, in this paper, we make use of a recently developed continuous scheme which is based on the concept of intrinsic dimensionality [5, 16]. In this concept, local image structures are organized continuously in a triangle. This approach is briefly described in section 3. Here, we show that the different classes of local image structures map to different distinguishable areas in the domain of the intrinsic dimensionality triangle (see figure 2) which is the first contribution of this paper.

To our knowledge, there does not exist a systematic and agreed classification of 3D local structures like there is for 2D local image structures (*i.e.*, homogeneous patches, edges, corners and textures). Intuitively, the 3D world consists of continuous surface patches and different kinds of 3D discontinuities. In the imaging process (through the lenses of camera or a retina), 2D local image structures are formed by these 3D structures together with the illumination and reflectivity of the environment.

With this intuition, any 3D scene can be decomposed geometrically into surfaces and 3D discontinuities. In this context, the local 3D structure of a point can be a:

- Surface Continuity: The underlying 3D structure can be described by one surface whose normal does not change or changes smoothly.
- Regular Gap discontinuity: The underlying 3D structure can be described by a small set of surfaces with a significant depth difference. The 2D and 3D views of an example gap discontinuity are shown in figure 1(a).
- Irregular Gap discontinuity: The underlying 3D structure shows high depth variation and can not be described by two or three surfaces. An example of an irregular gap discontinuity is shown in figure 1(b).
- Orientation Discontinuity: The underlying 3D structure can be described by two surfaces with significantly different 3D orientations that meet at the point whose 3D structure is being questioned. In this type of discontinuity, no gap but a change in 3D orientation between the meeting surfaces occurs. An example for this type of discontinuity is shown in figure 1(c).

3. Intrinsic Dimensionality

In image processing, intrinsic dimensionality was introduced by Zetsche and Barth[28] to distinguish between different local image structures. The idea is to assign intrinsically zero dimensionality (i0D), intrinsically one dimensionality (i1D) and intrinsically two dimensionality (i2D) to homogeneous patches, edges and corner-like structures, respectively. The concept of intrinsic dimensionality has been mostly applied in a discrete way which has been extended in [5, 16] to classify the local image structures continuously instead of giving them discrete labels.

In [5, 16], it has been also shown that the topological structure of the intrinsic dimensionality can be understood as a triangle whose corners correspond to the 'ideal' cases of 2D structures (*i.e.*, homogeneous patches, edges and corners). The inner of the triangle spans signals that carry aspects of the three 'ideal' cases, and the distance from the specific corners indicates the similarity (or dissimilarity) to the 'ideal' i0D, i1D and i2D signals. The horizontal and the vertical axes denote the contrast and the orientation variance, respectively. Contrast measures non-homogeneity whereas orientation variance measures the variation of orientation in a local patch describing the local image structure. An 'ideal' homogeneous image patch is expected to have zero contrast and zero orientation variance whereas an 'ideal' edge should have high contrast and zero orientation variance. An 'ideal' corner is supposed to have high contrast and high orientation variance.

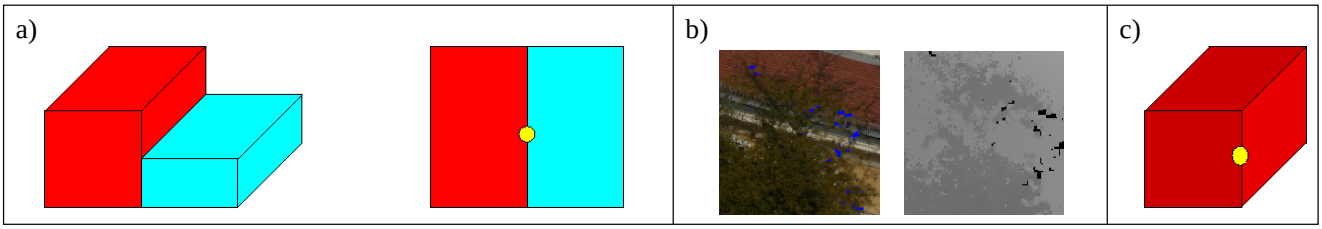


Figure 1. Examples for types of 3D discontinuities. Points of interest are marked with yellow circles. (a) 2D and 3D views of a gap discontinuity, (b) image (on the left) and range data (on the right) of an irregular gap discontinuity and (c) orientation discontinuity.

Figure 2 shows how the triangle of intrinsic dimensionality looks like and how a set of example local image structures map on to it. In figure 2, we see that different visual structures map to different areas in the triangle. A detailed analysis of how 2D structures are distributed over the intrinsic dimensionality triangle and how some visual information depends on this distribution can be found in [12]. Different from [12], in this paper, regarding this distribution, we show that textures also map to a different area of their own. The fact that different local image structures have their own distinguishable areas in the triangle provides us with a continuous classifier that distinguishes between homogeneous, edge-like, texture-like and corner-like structures.

4. Methods

In this section, we define our measures for the three kinds of discontinuities that we described in section 1; namely, gap discontinuity, irregular gap discontinuity and orientation discontinuity. The measures for gap discontinuity, irregular gap discontinuity and orientation discontinuity of a patch P will be respectively denoted by $\mu_{GD}(P)$, $\mu_{IGD}(P)$ and $\mu_{OD}(P)$. The reader who is not interested in the technical details can jump directly to section 5.

In our analysis, we used chromatic range data of outdoor scenes[1] which were obtained from Riegl UK Ltd. (`http://www.riegl.co.uk/`). There were 20 scenes in total, 10 of which are shown in figure 3. The range of an object which does not reflect the laser beam back to the scanner or is out of the range of the scanner cannot be measured. These points are marked with blue in figure 3 and are not processed in our analysis. The resolution range of the data set is [512-2048]x[390-2290] with an average resolution of 1140x1001.

3D discontinuities are detected in studies which involve range data processing, using different methods and using different names like two-dimensional discontinuous edge, jump edge or depth discontinuity for gap discontinuity; and, two-dimensional corner edge, crease edge or surface discontinuity for orientation discontinuity [1, 8, 25].

[1]We would like to note that it is problematic to do range scanning in nature scenes that include trees or other kinds of vegetation because of the unintended motion due to wind. As the image of the scene is taken after the scanning phase, this delay may make the image data fail to correspond to the range data.

4.1. Measure for Gap Discontinuity: μ_{GD}

Gap discontinuities can be measured or detected in a similar way to edges in 2D images; edge detection processes RGB-coded 2D images while for a gap discontinuity, one needs to process XYZ-coded 2D images. In other words, gap discontinuities can be measured or detected by taking a second order derivative of XYZ values [25].

Measurement of a gap discontinuity is expected to operate on both the horizontal and vertical axes of the 2D image; that is, it should be a two dimensional function. The alternative is to discard the topology and do 'edge-detection' in sorted XYZ values, *i.e.*, to operate as a one-dimensional function. Although we are not aware of a systematic comparison of the alternatives, for our analysis and for our data, the topology-discarding gap discontinuity measurement produced better results. Therefore, we have adopted the topology-discarding gap discontinuity measurement in the rest of the paper.

For an image patch P of size $N \times N$, let,

$$\begin{aligned} \mathcal{X} &= \text{ascending\_sort}(\{X_i \mid i \in P\}), \\ \mathcal{Y} &= \text{ascending\_sort}(\{Y_i \mid i \in P\}), \\ \mathcal{Z} &= \text{ascending\_sort}(\{Z_i \mid i \in P\}), \end{aligned} \tag{1}$$

and also, for $i = 1, .., (N \times N - 2)$,

$$\begin{aligned} \mathcal{X}^{\Delta} &= \{ \mid (\mathcal{X}_{i+2} - \mathcal{X}_{i+1}) - (\mathcal{X}_{i+1} - \mathcal{X}_i) \mid \}, \\ \mathcal{Y}^{\Delta} &= \{ \mid (\mathcal{Y}_{i+2} - \mathcal{Y}_{i+1}) - (\mathcal{Y}_{i+1} - \mathcal{Y}_i) \mid \}, \\ \mathcal{Z}^{\Delta} &= \{ \mid (\mathcal{Z}_{i+2} - \mathcal{Z}_{i+1}) - (\mathcal{Z}_{i+1} - \mathcal{Z}_i) \mid \}, \end{aligned} \tag{2}$$

where $\mathcal{X}_i, \mathcal{Y}_i, \mathcal{Z}_i$ represents 3D coordinates of pixel i.

The sets $\mathcal{X}^{\Delta}, \mathcal{Y}^{\Delta}$ and $\mathcal{Z}^{\Delta}$ are the measurements of the jumps (*i.e.*, second order differentials) in the sets $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$, respectively. A gap discontinuity can be defined simply as a measure of these jumps in these sets. In other words:

$$\mu_{GD}(P) = \frac{\phi(\mathcal{X}^{\Delta}) + \phi(\mathcal{Y}^{\Delta}) + \phi(\mathcal{Z}^{\Delta})}{3}, \tag{3}$$

where the function $\phi : \mathcal{S} \rightarrow [0, 1]$ over the set $\mathcal{S}$ measures the homogeneity of its argument set (in terms of its 'peakiness') and is defined as follows:

$$\phi(\mathcal{S}) = \frac{1}{\#(\mathcal{S})} \times \sum_{i \in \mathcal{S}} \frac{s_i}{\max(\mathcal{S})}, \tag{4}$$

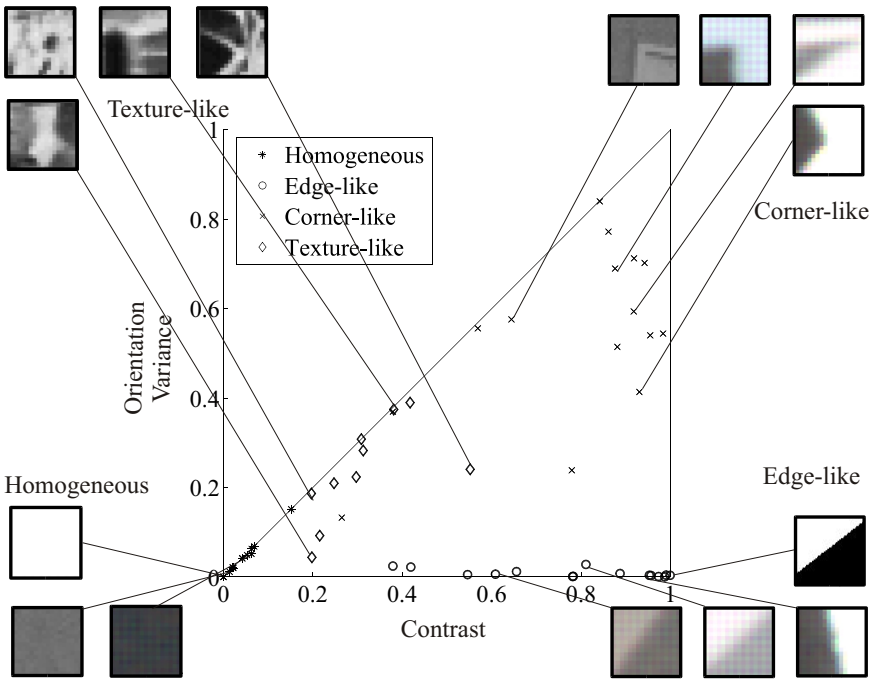


Figure 2. How a set of 54 patches map to the different areas of the intrinsic dimensionality triangle. Some examples from these patches are also shown. The horizontal and vertical axes of the triangle denote the contrast and the orientation variances of the image patches, respectively.

Figure 3. 10 of the 20 3D data sets used in the analysis. The points that don't have range data are marked in blue. The gray image shows the range data of the top-left scene. The resolution range is [512-2048]x[390-2290] with an average resolution of 1140x1001.

where $\#(\mathcal{S})$ is the number of the elements of $\mathcal{S}$, and s_i is the i^{th} element of the set $\mathcal{S}$. Note that as a homogeneous set (*i.e.*, a non-gap discontinuity) $\mathcal{S}$ produces a high $\phi(\mathcal{S})$ value, a gap discontinuity causes a low μ_{GD} value. Figure 5(c) shows the performance of μ_{GD} on one of our scenes shown in figure 3.

4.2. Measure for Orientation Discontinuity: μ_{OD}

The orientation discontinuity of a patch P can be detected or measured by taking the 3D orientation difference of the surfaces which meet at P. As the size of the patch P is small enough, the surfaces can be, in practice, approximated by 2-pixel wide unit planes. The histogram of the 3D orientation differences between every pair of unit planes forms one cluster for continuous surfaces and two clusters for orientation discontinuities.

For an image patch P of size $N \times N$ pixels, the orientation discontinuity measure is defined as:

$$\mu_{\mathbf{OD}}(P) = \psi(H^n(\{\alpha(i,j) \mid i,j \in planes(P), i \neq j\})), \quad (5)$$

where $H^n(S)$ is a function which computes the n-bin histogram of its argument set $\mathcal{S}$; $\psi(\mathcal{S})$ is a function which finds the number of clusters in $\mathcal{S}$; $planes(P)$ is a function which fits 2-pixel-wide unit planes to 1-pixel apart points in P using Singular Value Decomposition[2]; and, $\alpha(i,j)$ is the angle between planes i and j.

For a histogram H of size N_H, the number of clusters is:

$$\psi(S) = \frac{\sum_{i=1}^{N_H+1}(H_i > \frac{\max(H)}{10}) \neq (H_{i-1} > \frac{\max(H)}{10})}{2}, \quad (6)$$

[2]Singular Value Decomposition is a standard technique for fitting planes to a set of points. It finds the perfectly fitting plane if it exists; otherwise, it returns the least-square solution.

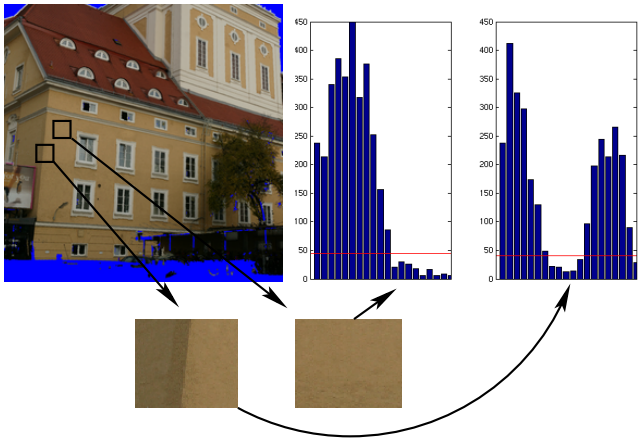


Figure 4. Example histograms and the number of clusters that the function $\psi(S)$ computes. $\psi(S)$ finds one cluster in the left histogram and two clusters in the right histogram. Red line marks the threshold value of the function. X axis denotes the values for 3D orientation differences.

where the operator $\neq$ returns 1 if its operands are not equal and returns 0, otherwise; H_i represents the i^{th} element of the histogram H; H_0 and H_{N_H+1} are defined as zero; and, $\max(H)/10$ is an empirical value which functions as the threshold value for finding the clusters. Figure 4 shows two example clusters for a continuous surface and an orientation discontinuity. Figure 5(d) shows the performance of μ_{OD} on one of our scenes shown in figure 3.

4.3. Measure for Irregular Gap Discontinuity: μ_{IGD}

Irregular gap discontinuity of a patch P can be measured by making use of the observation that an irregular-gap discontinuous patch from nature usually consists of small surface fragments with different 3D orientations. Therefore, the amount of variety in the 3D orientation histogram of a patch P can measure the irregular gap discontinuity of P.

Similar to the measure for orientation discontinuity defined in section 4.2, the histogram of the differences between the 3D orientations of the unit planes (which are of 2 pixels wide) is analyzed. For an image patch P of size $N \times N$ pixels, the irregular gap discontinuity measure is defined as:

$$\mu_{\mathbf{IGD}}(P) = \phi(H^n(\{\alpha(i,j) \mid i,j \in planes(P), i \neq j\})), \qquad (7)$$

where $planes(P)$, $\alpha(i,j)$, $H^n(\mathcal{S})$ and $\phi(\mathcal{S})$ are as defined in section 4.2. Figure 5(e) shows the performance of μ_{IGD} on one of our scenes shown in figure 3.

The relation between the measurements and the types of the 3D discontinuities are outlined in table 1 which entails that an image patch P is:

- gap discontinuous if $\mu_{GD}(P) < T_g$ and $\mu_{IGD}(P) < T_{ig}$,
- irregular-gap discontinuous if $\mu_{GD}(P) < T_g$ and $\mu_{IGD}(P) > T_{ig}$,
- orientation discontinuous if $\mu_{GD}(P) \geq T_g$ and $\mu_{OD} > 1$,

Dis. Type	μ_{GD}	μ_{IGD}	μ_{OD}
Continuity	High value	Don't care	1
Gap Dis.	Low value	Low value	Don't care
Irregular Gap Dis.	Low value	High value	Don't care
Orientation Dis.	High value	Don't care	> 1

Table 1. The relation between the measurements and the types of the 3D discontinuities.

- continuous if $\mu_{GD}(P) \geq T_g$ and $\mu_{OD}(P) \leq 1$.

For our analysis, we have taken N and the threshold values T_g, T_{ig} empirically as 10, 0.4 and 0.6, respectively. The number of bins, n, in H^n is taken as 20.

Figure 5(a) shows the types of 3D discontinuities marked in four different colors for every pixel of the scenes shown in figure 3. We see that our measures can capture the 3D structure of the data sufficiently correct.

5. Results and Discussion

For each pixel of the scene (except for pixels where range data is not available), we computed the 3D discontinuity type and the intrinsic dimensionality. Figure 5(a) and (b) shows the images where the 3D discontinuity and the intrinsic dimensionality of each pixel are marked with different colors.

Having the 3D discontinuity type and the information about the local 2D structure of each point, it is straightforward to compute the probability $P(\text{3D Discontinuity} \mid \text{2D Structure})$, which is shown in figure 6. Note that the four triangles in figures 6(a), 6(b), 6(c) and 6(d) add up to one for all points of the triangle. We see that:

- Figure 6(a) shows that homogeneous image patches correspond to 3D continuities.

  Many surface reconstruction studies make use of a basic assumption that there is a smooth surface between any two points in the 3D world, if there is no contrast difference between these points in the image. This assumption has been first called as 'no news is good news' in [6]. With figure 6(a), we quantify 'no news is good news' and show for which structures and to what extent it holds. In addition to the fact that no news is in fact good news, the figure shows that news, especially texture-like structures and edge-like structures, can also be good news (see below).
- Edges are considered as important sources of information for object recognition and reliable correspondence finding. Approximately 10% of local image structures are of that type (see, *e.g.*, [12]). Figures 6(a), (b) and (d) show that most of the edges correspond to continuous surfaces or gap discontinuities. The edges that correspond to continuous surfaces are mostly low-contrast edges. Little percentage of the edges are formed by orientation discontinuities.

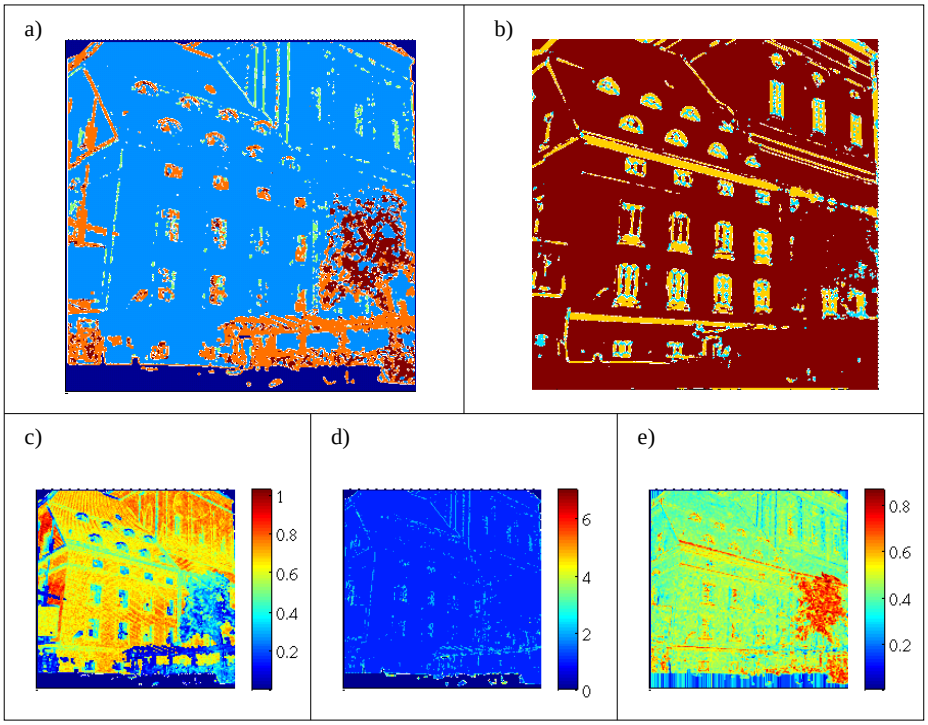


Figure 5. The 3D and 2D information for one of the scenes shown in figure 3. Dark blue marks the points without range data. (a) 3D discontinuity. Blue: continuous surfaces, light blue: orientation discontinuities, orange: gap discontinuities and brown: irregular gap discontinuities. (b) Intrinsic Dimensionality. Homogeneous patches, edge-like and corner-like structures are encoded in colors brown, yellow and light blue, respectively. (c) Gap discontinuity measure μ_{GD}. (d) Orientation discontinuity measure μ_{OD}. (e) Irregular gap discontinuity measure μ_{IGD}.

- Figure 6(b) shows that well-defined corner-like structures result from either gap discontinuities or continuities.
- Textures also map with high likelihood to surface continuities but also to irregular gap discontinuities.

  Finding correspondences becomes more difficult with the lack or repetitiveness of the local structure. The estimates of the correspondences at texture-like structures are naturally less reliable. In this sense, the likelihood that certain textures are caused by continuous surfaces (shown in figure 6(a)) can be used to model stereo matching functions that include interpolation as well as information about possible correspondences based on the local image information.

It is remarkable that local image structures mapping to different sub-regions in the triangle are caused by rather different 3D structures. This clearly indicates that these different image structures should be used in different ways for surface reconstruction.

6. Conclusion

In this paper, using 3D range data with real-world color information, we have analyzed the conditional probability of a 3D structure given the 2D structure. With this probability, we could investigate the relation between 2D structures and the underlying 3D structures as well as analyze the validity of a widely-used assumption/smoothing constraint, namely, 'no news is good news' [6].

Besides, we have presented a continuous classification scheme which can be used to distinguish between homogeneous, edge-like, corner-like and texture-like structures. By taking a higher-order representation than existing range-data analysis studies, we could point to the intrinsic properties of the 3D world and its relation to the image data. This analysis is important because (1) it may be that the human visual system is adapted to the statistics of the environment [2, 13, 15, 18, 21, 22], and (2) it may be used in several computer vision applications like depth estimation in a similar way as in [3, 4, 20, 29].

In our current work, the probability distributions will be used for estimating the 3D structure from 2D structure in a Bayesian framework for surface reconstruction/interpolation studies.

7. Acknowledgments

We would like to thank RIEGL UK Ltd. for providing us with 3D range data. This work is supported by the ECO-

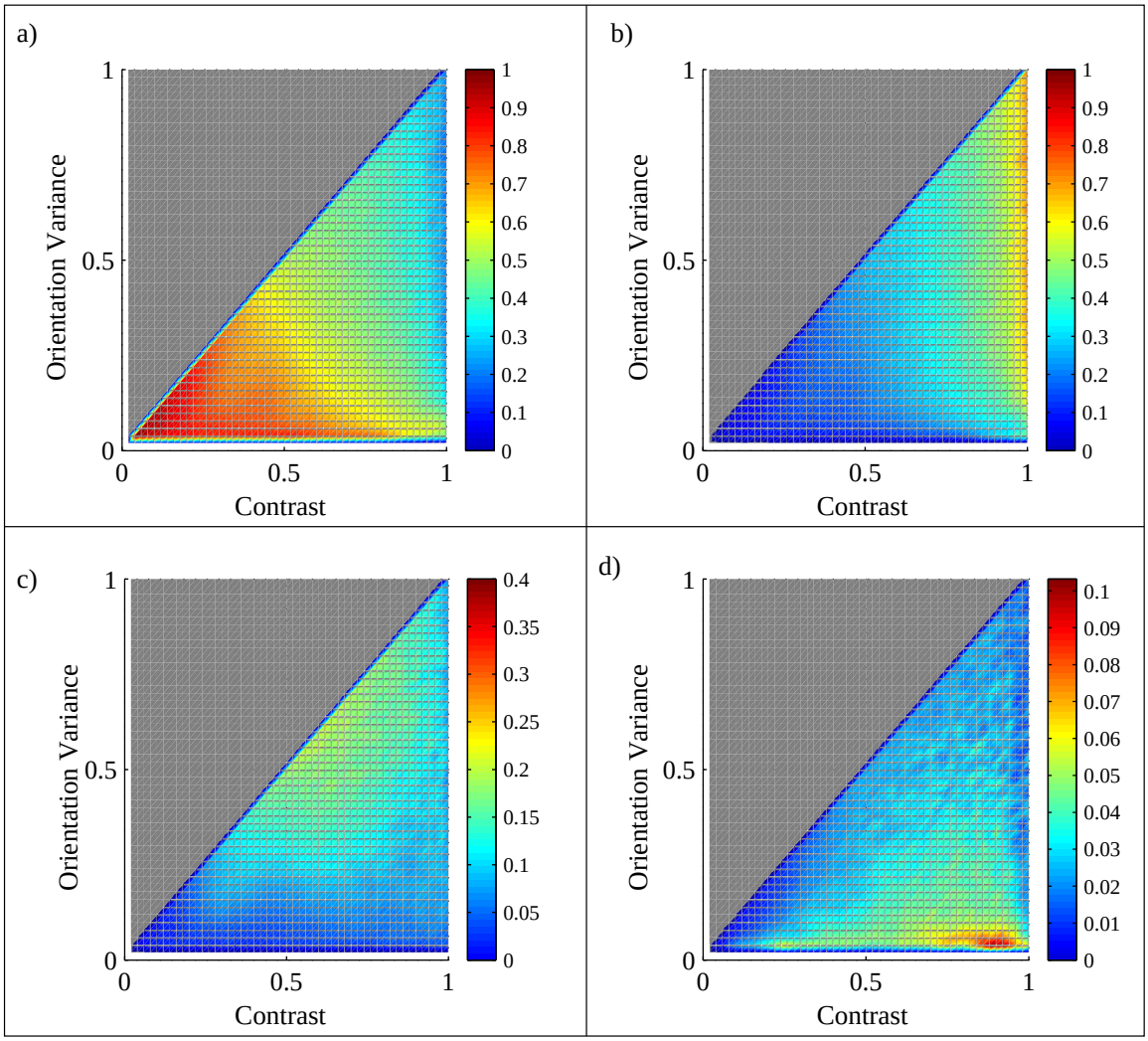


Figure 6. P(3D Discontinuity | 2D Structure): (a) P(Continuity | 2D Structure). (b) P(Gap Discontinuity | 2D Structure). (c) P(Irregular Gap Discontinuity | 2D Structure). (d) P(Orientation Discontinuity | 2D Structure).

VISION project.

References

[1] R. M. Bolle and B. C. Vemuri. On three-dimensional surface reconstruction methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1):1–13, 1991.

[2] E. Brunswik and J. Kamiya. Ecological cue–validity of 'proximity' and of other Gestalt factors. *American Journal of Psychologie*, LXVI:20–32, 1953.

[3] H. Elder and R. Goldberg. Ecological statistics of gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2(4):324–353, 2002.

[4] J. H. Elder, A. Krupnik, and L. A. Johnston. Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(25):1–14, 2003.

[5] M. Felsberg and N. Krüger. A probablistic definition of intrinsic dimensionality for images. *Pattern Recognition, 24th DAGM Symposium*, 2003.

[6] W. E. L. Grimson. Surface consistency constraints in vision. *Computer Vision, Graphics and Image Processing*, 24(1):28–51, Oct. 1983.

[7] A. Guzman. Decomposition of a visual scene into three-dimensional bodies. *AFIPS Fall Joint Conference Proceedings*, 33:291–304, 1968.

[8] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. Bowyer, D. W. Eggert, A. Fitzgibbon, and R. B. Fisher. An experimental comparison of range image segmentation algorithms.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(7):673–689, 1996.

[9] C. Q. Howe and D. Purves. Range image statistics can explain the anomalous perception of length. *PNAS*, 99(20):13184–13188, 2002.

[10] C. Q. Howe and D. Purves. Size contrast and assimilation explained by the statistics of natural scene geometry. *Journal of Cognitive Neuroscience*, 16(1):90–102, 2004.

[11] J. Huang, A. B. Lee, and D. Mumford. Statistics of range images. *CVPR*, 1(1):1324–1331, 2000.

[12] S. Kalkan, D. Calow, F. Wörgötter, M. Lappe, and N. Krüger. Local image structures and optic flow estimation. *Accepted for Network: Computation in Neural Systems*, 2005.

[13] D. C. Knill and W. Richards, editors. *Perception as bayesian inference*. Cambridge: Cambridge University Press, 1996.

[14] J. Koenderink and A. Dorn. The shape of smooth objects and the way contours end. *Perception*, 11:129—173, 1982.

[15] N. Krüger. Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129, 1998.

[16] N. Krüger and M. Felsberg. A continuous formulation of intrinsic dimension. *Proceedings of the British Machine Vision Conference*, 2003.

[17] D. Marr. *Vision: A computational investigation into the human representation and processing of visual information*. Feeman, 1977.

[18] B. Olshausen and D. Field. Natural image statistics and efficient coding. *Network*, 7:333–339, 1996.

[19] B. Potetz and T. S. Lee. Statistical correlations between two-dimensional images and three-dimensional structures in natural scenes. *Journal of the Optical Society of America*, 20(7):1292–1303, 2003.

[20] N. Pugeault, N. Krüger, and F. Wörgötter. A non-local stereo similarity based on collinear groups. *Proceedings of the Fourth International ICSC Symposium on Engineering of Intelligent Systems*, 2004.

[21] D. Purves and B. Lotto, editors. *Why we see what we do: an empirical theory of vision*. Sunderland, MA: Sinauer Associates, 2002.

[22] R. P. N. Rao, B. A. Olshausen, and M. S. Lewicki, editors. *Probabilistic models of the brain*. MA: MIT Press, 2002.

[23] N. Rubin. The role of junctions in surface completion and contour matching. *Perception*, 30:339–366, 2001.

[24] I. A. Shevelev, V. M. Kamenkovich, and G. A. Sharaev. The role of lines and corners of geometric figures in recognition performance. *Acta Neurobiol Exp*, 63(4):361–368, 2003.

[25] Y. Shirai. *Three-dimensional computer vision*. Springer-Verlag New York, Inc., 1987.

[26] M. Tuceryan and N. K. Jain. Texture analysis. *The Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, pages 207–248, 1998.

[27] Z. Yang and D. Purves. Image/source statistics of surfaces in natural scenes. *Network: Computation in Neural Systems*, 14:371–390, 2003.

[28] C. Zetzsche and E. Barth. Fundamental limits of linear filters in the visual processing of two dimensional signals. *Vision Research*, 30(7):1111–1117, 1990.

[29] S. C. Zhu. Embedding gestalt laws in markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1170–1187, 1999.

Robotics Group
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Technical Report no. 2007 – 2

Depth Prediction at Homogeneous Image structures

Sinan Kalkan, Florentin Wörgötter, Norbert Krüger

January 22, 2007

Title	Depth Prediction at Homogeneous Image structures
Author(s)	Sinan Kalkan, Florentin Wörgötter, Norbert Krüger
Publication History	

Abstract

Depth at homogeneous or weakly-textured image areas is difficult to obtain because such image areas suffer the well-known correspondence problem. In this paper, we propose a voting model that predicts the depth at such image areas from the depth of bounding edge-like structures. The depth at edge-like structures is computed using a feature-based stereo algorithm, and is used to vote for the depth of homogeneous image areas. We show the results of our ongoing work on different scenarios.

1 Introduction

Extraction of 3D structure from 2D images is realized utilizing a set of inverse problems that include structure from motion, stereo vision, shape from shading, linear perspective, texture gradients and occlusion [3]. These cues can be classified as pictorial, or monocular (such as shading, utilization of texture gradients or linear perspective) and multi-view (like stereo and structure from motion). Depth cues which make use of multiple views require correspondences between different 2D views of the scene. In contrast, pictorial cues use statistical and geometrical relations in one image to make statements about the underlying 3D structure. Many surfaces have only weak texture or no texture at all, and as a consequence, the *correspondence problem is very hard or not at all resolvable for these surfaces*. Nevertheless, humans are able to reconstruct 3D information for these surfaces, too. Existing psychophysical experiments (see, *e.g.*, [2, 4]) and computational theories (see, *e.g.*, [1, 6, 24]) suggest that in the human visual system, *an interpolation process* is realized that starting with the local analysis of edges, corners and textures, computes depth also in areas where correspondences cannot easily be found.

In this paper, we are interested in prediction of depth at homogeneous image patches (called *mono*s in this paper) from the depth of the edges in the scene using a voting model. We start by creating a representation of the input stereo images in terms of local image patches corresponding to edge-like structures and monos (as introduced in [14] and section 2, and described in detail in [15]). The depth at edge-like patches is extracted using feature-based stereo computation between the two images (using the method introduced in [20]). The depth that is extracted at the bounding edge-like patches of a mono using stereo votes for its depth.

We would like to distinguish *depth prediction* from *surface interpolation* because surface interpolation assumes that there is already a dense depth map of the scene available in order to be able to estimate the 3D orientation at points (see, *e.g.*, [6, 7, 8, 17, 18, 23, 24]) whereas our understanding of depth prediction makes use of only 3D line-orientations at edge-segments which are computed using a feature-based stereo proposed in [20].

(a) (b) (c)

Figure 1: An input stereo pair ((a) and (b)) and how a feature-based stereo algorithm (taken from [20]) looks like (c).

A typical scenario that our model is designed for is shown in figure 1 where an input stereo pair and the stereo data (computed using [20]) are displayed. We see that computed stereo information has strong outliers which prohibit a *surface interpolation* method as it is not possible to differentiate between the outliers and the reliable stereo information. Moreover, the stereo information that should be reliable at the edges of the road turn out not to share a common surface nor the same 3D line (see figure 1(c)). Applying a surface interpolation method on such input data is expected to lead to a wrong road surface prediction. In this paper, we will show that our depth prediction method is able to cope with such strong outliers.

1.1 Related studies

It is fair to count the early works of Grimson [6] as the pioneers of surface interpolation. In [6], Grimson proposed fitting square Laplacian functionals to surface orientations at existing 3D points utilizing a *surface consistency constraint* called 'no news is good news'. The constraint argues that if two image points do not have a contrast difference in-between, then they can be assumed to be on the same 3D surface (see [11] for a quantification of this assumption). This work is extended in [7] with use of shading information. [6, 7] assume that surface information is available, and the input 3D points are dense enough for second order differentiation.

In [1], surface orientation at homogeneous image areas is recovered by *interpreting* line *drawings*. Lines are classified as extremal or discontinuity by making use of the junction labels and global relations like symmetry and parallellism. They assume that (1) extremal points (the boundaries of the objects) in an image correspond to surface orientations which are normal to the image curve and the line of sight, and that (2) discontinuities (lines other than extremal points) lead to surface orientations which are normal to space curve. The underlying assumptions of [1] are that (1) a clean contour of the scene is provided, and that (2) the object is separated from the background. Moreover, the results provided in [11] suggest that it may not be a good idea to assume that edges correspond to only certain types of surface orientations. [19, 22, 25, 26] are similar to [1] as far as our paper is concerned.

In [8], 3D points with surface orientation are interpolated using a perceptual constraint called *co-surfacity* which produces a 3D association field (which is called Diabolo field by the authors) similar to the association field used in 2D perceptual contour grouping studies. If the points do not have 3D orientation, they estimate the 3D orientation first and then apply the surface interpolation step. In [17, 18], it is argued that stereo matching and surface interpolation should not be sequential but rather simultaneous. For this, they employ the following steps: (1) Normalized-cross correlation and edge-based stereo are computed. (2) The disparities are combined and disparities corresponding to inliers, surfaces and surface discontinuities are marked using tensor voting. (3) Surfaces are extracted using marching cubes approach. At this stage, surfaces are over the boundaries. (4) At the last step, over-boundary surfaces are trimmed. They assume sphere as their surface model when interpolating surface orientations.

In [23, 24], stereo is computed at different scales, and instead of collapsing the results of these different scales into a single layer of disparity estimation and then applying surface interpolation, surface interpolation is applied separately for each scale and the results are combined.

Our work is different from the above mentioned worksin that:

- Our approach does not assume that the input stereo points are dense enough to compute their 3D orientation (this is why the authors of this paper prefer to distinguish between depth prediction and surface interpolation). Instead, our method relies on the 3D line-orientations of the edge segments which are extracted using a feature-based stereo algorithm (proposed in [20]).

- We employ a voting method like [17, 18] but is different, allowing long-range interactions in empty image areas, in order to predict *both* the depth and the surface orientation.

The paper is organized as follows: In section 2, we introduce how the images are represented in terms of local image patches. Section 3 describes the 2D and 3D relations between the local image patches that are utilized in the depth prediction process. Section 4 gives the outline of how the depth prediction is performed. In section 5, the results are presented and discussed. Finally, in section 6, the paper is concluded.

2 Visual Features

The visual features we utilize (called primitives in the rest of the paper) are local, multi-modal feature descriptors that were intoduced in [14]. They are semantically and geometrically meaningful descriptions of local patches, motivated by the hyper-columnar structures in V1 ([9]).
An edge-like primitive can be formulated as:

$$\pi^{\mathrm{e}} = (\mathbf{x}, \theta, \omega, (\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r), f), \tag{1}$$

where $\mathbf{x}$ is the image position of the primitive; θ is the 2D orientation; ω represents the contrast transition; $(\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r)$ is the representation of the color, corresponding to the left ($\mathbf{c}_l$), the middle ($\mathbf{c}_m$) and the right side ($\mathbf{c}_r$) of the primitive; and, f is the optical flow extracted using Nagel-Enkelmann optic flow algorithm.
As the underlying structure of an homogeneous image patch is different from that of an edge-like patch, a different representation is needed for homogeneous image structures (called *mono*s in this paper):

$$\pi^{\mathrm{m}} = (\mathbf{x}, \mathbf{c}), \tag{2}$$

where $\mathbf{x}$ is the image position, and $\mathbf{c}$ is the color of the mono.
See [16] for more information about these modalities and their extraction. Figure 2 shows extracted primitives for an example scene.
π^{e} is a 2D feature which can be used to find correspondences in a stereo framework to create 3D primitives (as introduced in [13, 21]) with the following formulation:

$$\Pi^{\mathrm{e}} = (\mathbf{X}, \Theta, \Omega, (\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r)), \tag{3}$$

where $\mathbf{X}$ is the 3D position; Θ is the 3D orientation; Ω is the phase (i.e., contrast transition); and, $(\mathbf{c}_l, \mathbf{c}_m, \mathbf{c}_r)$ is the representation of the color, corresponding to the left ($\mathbf{c}_l$), the middle ($\mathbf{c}_m$) and the right side ($\mathbf{c}_r$) of the 3D primitive.
In this paper, we estimate the 3D representation Π^{m} of monos which stereo fails to compute:

$$\Pi^{\mathrm{m}} = (\mathbf{X}, \mathbf{n}, \mathbf{c}), \tag{4}$$

where $\mathbf{X}$ and $\mathbf{c}$ are as in equation 2, and $\mathbf{n}$ is the orientation (*i.e.*, normal) of the plane that locally represents the mono.

3 Relations between Primitives

Sparse and symbolic nature of primitives allows the following relations to be defined on them. For more information about relations of primitives, see [10].

(a) Input image.

(b) Extracted primitives.

Figure 2: Extracted primitives (b) for the example image in (a).

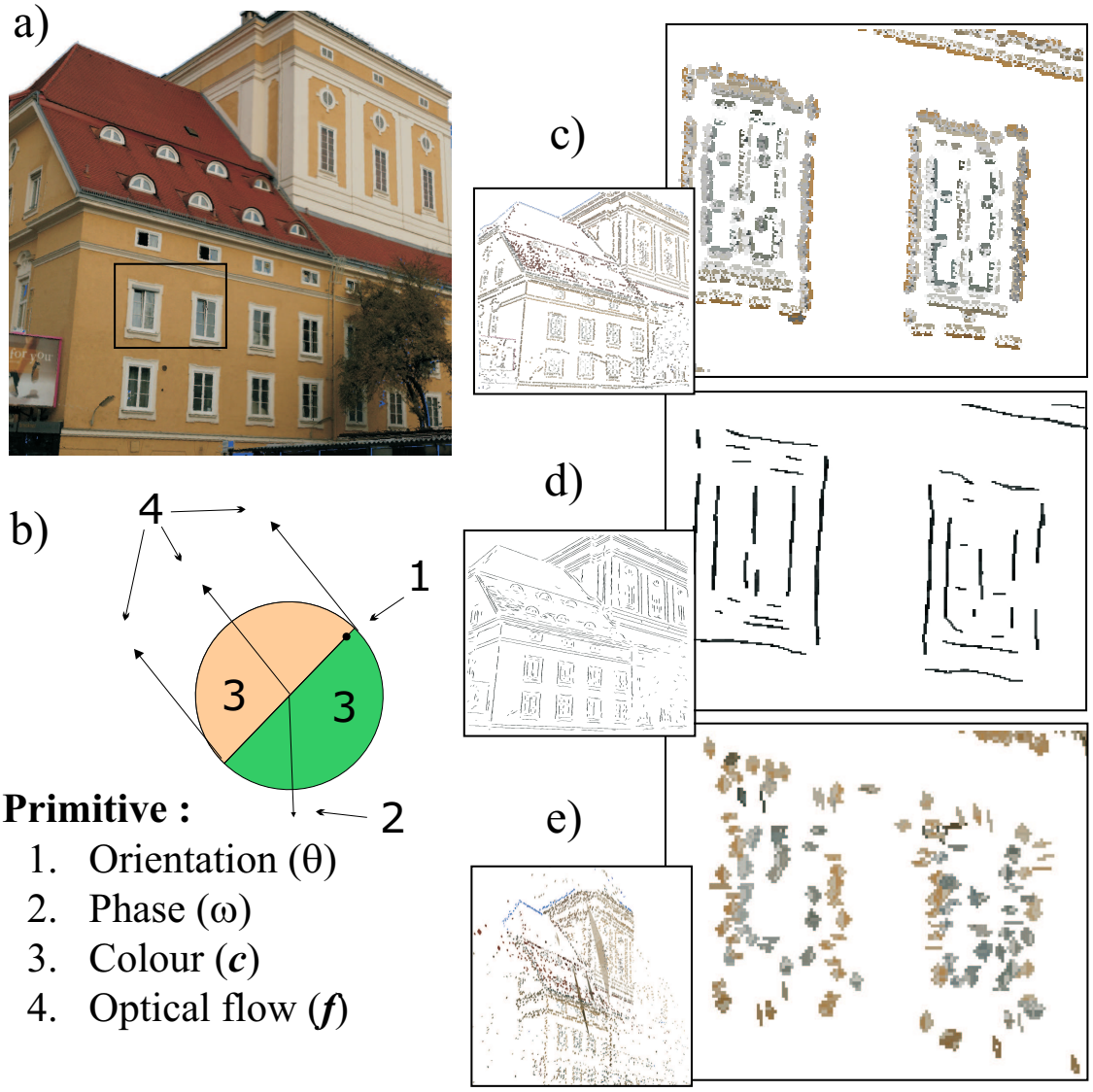


Figure 3: Illustration of the primitive extraction process from a video sequence. The 2D–primitives extracted from the input image (a) (see section 2), and finally the 3D–primitives reconstructed from the stereo–matches as described as described in [21]. **(a)** An example input image. **(b)** A graphic description of the 2D–primitives. **(c)** A magnification of the image representation. **(d)** Perceptual grouping of the primitives as described in [21]. **(e)** The reconstructed 3D entities. Note that the structure reconstructed is quite far from the cameras, leading to a certain imprecision in the reconstruction of the 3D–primitives. A simple scheme addressing this problem is described in [21].

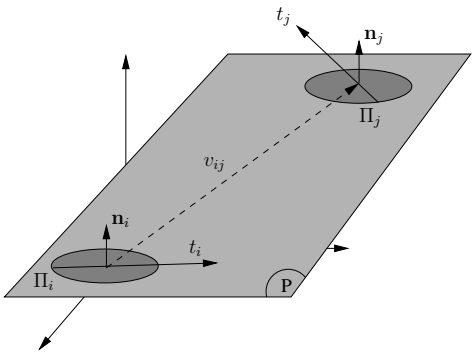


Figure 4: Co–planarity of two 3D primitives Π_i^e and Π_j^e.

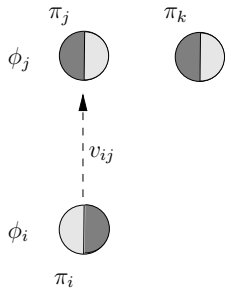


Figure 5: Linear dependence of three $\pi^{\mathrm{e}}{}_i$, $\pi^{\mathrm{e}}{}_j$ and $\pi^{\mathrm{e}}{}_k$. In this example, $\pi^{\mathrm{e}}{}_i$ is linearly dependent with $\pi^{\mathrm{e}}{}_j$ whereas $\pi^{\mathrm{e}}{}_k$ is linearly independent of other primitives.

3.1 Co–planarity

Two 3D edge primitives Π_i^e and Π_j^e are co–planar iff their orientation vectors lie on the same plane, i.e.:

$$cop(\Pi_i^e, \Pi_j^e) = 1 - |\mathbf{proj}_{t_j \times v_{ij}}(t_i \times v_{ij})|, \tag{5}$$

where v_{ij} is defined as the vector $(\mathbf{X}_i - \mathbf{X}_j)$; t_i and t_j denote the vectors defined by the 3D orientations Θ_i and Θ_j, respectively; and, $\mathbf{proj}_{\mathbf{u}}(\mathbf{a})$ is defined as:

$$\mathbf{proj}_{\mathbf{u}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{u}}{\| \mathbf{u} \|^2}\mathbf{u}. \tag{6}$$

The co–planarity relation is illustrated in Fig. 4.

3.2 Linear dependence

Two 3D primitives Π_i^e and Π_j^e are linearly dependent iff the *three* lines which are defined by (1) the 3D orientation of Π_i^e, (2) the 3D orientation of $\Pi^{\mathrm{e}}{}_j$ and (3) v_{ij} are identical. Due to uncertainty in the 3D reconstruction process, in this work, the linear dependence of two spatial primitives Π_i^e and Π_j^e is computed using their 2D projections π_i^e and π_j^e. We define the linear dependence of two 2D primitives π_i^e and π_j^e as:

$$lin(\pi_i^e, \pi_j^e) = |\mathbf{proj}_{v_{ij}} t_i| > Th \ \wedge \ |\mathbf{proj}_{v_{ij}} t_j| > Th, \tag{7}$$

where t_i and t_j are the vectors defined by the orientations θ_i and θ_j, respectively; and, Th is a threshold.

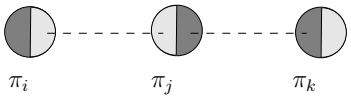


Figure 6: Co–colority of three 2D primitives π_i^e, π_j^e and π_k. In this example, π_i^e and π_j^e are cocolor, so are π_i^e and π_k^e; however, π_j^e and π_k^e are not cocolor.

3.3 Co–colority

Two 3D primitives Π_i^e and Π_j^e are co–color iff their parts that face each other have the same color. In the same way as linear dependence, co–colority of two spatial primitives Π_i^e and Π_j^e is computed using their 2D projections π_i^e and π_j^e. We define the co–colority of two 2D primitives π_i^e and π_j^e as:

$$coc(\pi_i^e, \pi_j^e) = 1 - \mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j), \tag{8}$$

where $\mathbf{c}_i$ and $\mathbf{c}_j$ are the RGB representation of the colors of the parts of the primitives π_i^e and π_j^e that face each other; and, $\mathbf{d}_c(\mathbf{c}_i, \mathbf{c}_j)$ is Euclidean distance between RGB values of the colors $\mathbf{c}_i$ and $\mathbf{c}_j$.
Co-colority between an edge primitive π^{e} and and a mono primitive π^{m}, and between two monos can be defined similarly (not shown here).
In Fig. 6, a pair of co–color and not co–color primitives are shown.

4 Formulation of the model

For the prediction of the depth at monos, we developed a voting model. In a voting model, there are a set of voters that state their *opinion* about a certain event e. A voting model combines these votes in a reasonable way to make a decision about the event e.
In the depth prediction problem, the event e to be voted about is the depth and the 3D orientation of a mono π^{m}, and the voters are the edge primitives $\{\pi_i^e\}$ (for $i = 1, ..., N_E$) that bound the mono. In this paper, we are interested in the predictions of pairs of π_i^es, which are denoted by P_j for $j = 1, ..., N_P$. While forming a pair P_j from two edges π_i^e and π_k^e from the set of the bounding edges of a mono π^{m}, we have the following restrictions:

1. π_i^e and π_k^e should share the same color with the mono π^{m} (*i.e.*, the following relations should hold: $coc(\pi_i^e, \pi_k^e)$ and $coc(\pi_i^e, \pi^{\mathrm{m}})$).

2. The 3D primitives Π_i^e and Π_k^e of π_i^e and π_k^e should be on the same plane (*i.e.*, $cop(\Pi_i^e, \Pi_k^e)$).

3. π_i^e and π_k^e should not be linearly dependent so that they can define only one plane (*i.e.*, $\neg\, lin(\pi_i^e, \pi_k^e)$).

In figure 7, such restrictions are illustrated for an example mono and a set of edge primitives that bound it. The primitives π_j^e and π_m^e are on the same line (*i.e.*, they are linearly dependent), and they define infinitely many planes. As for primitives π_l^e and π_k^e, they cannot define a plane as they are not on the same plane, nor do they share the same color.
The vote v_i by a pair P_j can be parametrized by:

$$v_i = (\mathbf{X}, \mathbf{n}), \tag{9}$$

where $\vec{n}$ is the normal of the mono π^{m}, and z is its depth relative to the plane defined by P_i.

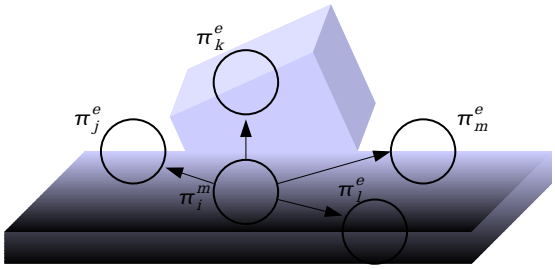


Figure 7: A set of primitives for illustrating why the relations coplanarity, cocolority and linear dependence are required as restrictions for forming pairs from edges.

Each v_i has an associated reliability or probability r_i. They denote how likely the vote is based on the believes of pair P_i. It can be modeled as a function of the distance of the mono π^{m} to the intersection point IP:

$$r_i = f(d(\Pi^{\mathrm{m}}, P_i)). \tag{10}$$

r_i can be weighted by the confidences of the elements of the pair P_i that reflect their quality.

4.1 Bounding edges of a mono

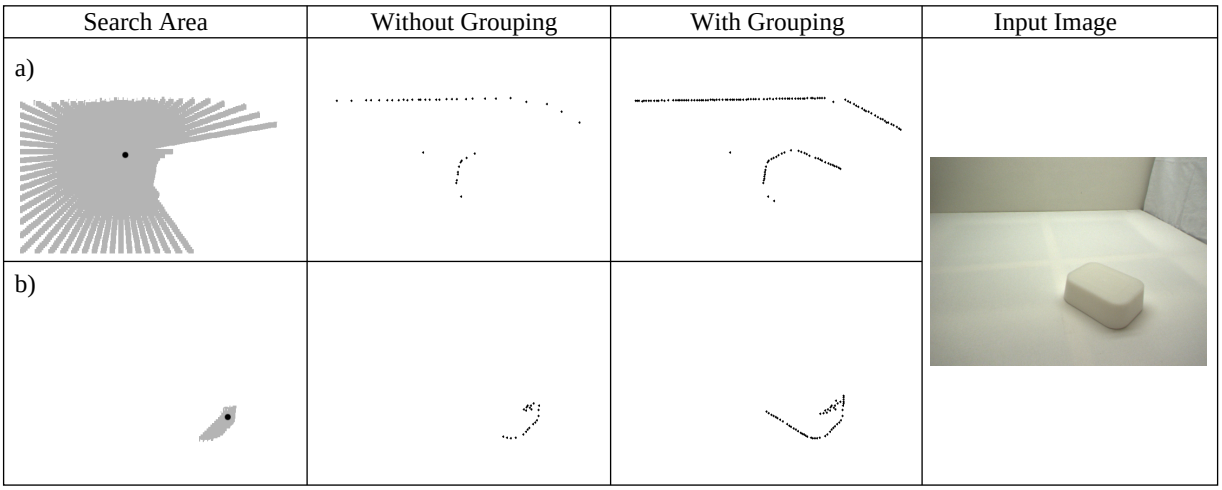


Figure 8: Finding bounding edge primitives with and without grouping information for two different monos which are marked in black in the first column. Using grouping information produces a more complete boundary finding as shown in (a). However, using grouping may include unwanted edge primitives in the boundary as shown in (b).

Finding the bounding edges of a mono π^{m} requires making searches in a set of directions d_i, $i = 1...N_d$ for the edge primitives. In each direction d_i, starting from a minimum distance R_{min}, the search is performed upto a distance of R_{max} in discrete steps s_j, $j = 1...N_s$. If an edge primitive π^{e} is found in direction d_i in the neighborhood Ω of a step s_j, π^{e} is added to the list of bounding edges and the search continues with the next direction.
The above mentioned method for finding the bounding edge primitives will lead to an incomplete and sparse boundary detection (see figure 8) because the search is performed only in a set of discrete directions. This can be improved by making use of the contour grouping information; when an edge primitive π^{e} is found

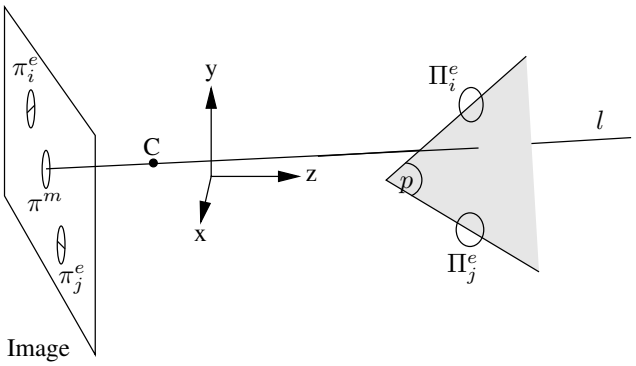


Figure 9: Illustration of how the vote of a pair of edge primitives is computed. The 3D primitives Π_i^e and Π_j^e corresponding to the 2D primitives π_i^e and π_j^e define the plane p. The intersection of p with the ray l that goes through the 2D mono π^{m} and the camera center C then determines the position of the estimated 3D mono Π^{m}. The 3D orientation of Π^{m} is set to be the orientation of the plane p.

in a direction d_i at step s_j, if π^{e} is part of a group G, then all the edge primitives in G can be added to the list of bounding edges (see [21] for information about the grouping method we employ in this paper).
Grouping information can lead to more complete and dense boundary finding as shown in figure 8(a); however, for certain objects, it may lead to worse results due to low contrast edges (see figure 8(b)).

4.2 The vote of a pair of edge primitives on a mono π^{m}

A pair P_i of two edge primitives π_j^e and π_k^e with two corresponding 3D edge primitives Π_j^e and Π_k^e, which are co-planar, co-color and linearly *independent*, defines a plane p with 3D normal $\mathbf{n}$ and position $\mathbf{X}$.
The vote v_l of Π_j^e and Π_k^e is computed by the intersection of the plane p with the ray l that goes through the mono, π^{m}, and the focus of the camera (see figure 9). The ray l is computed using the following formula ([5], pg41):

$$\mathrm{X_a} = P^{-1}(-\tilde{p} + \lambda\tilde{x}), \tag{11}$$

where $\tilde{x}$ is the homogeneous position of π^{m}; P and $\tilde{p}$ are respectively the 3x3 and the 3x1 sub-parts of the 3x4 projection matrix P_m so that $P_m = [P\ \ \tilde{p}]$; and, λ is an arbitrary number. By using two different values for λ, two different points on ray l are extracted which then are used to compute the ray l.
Because the ray l is unique for a mono π^{m}, all the votes processed for the mono π^{m} will be on ray l. This property can be exploited for clustering the votes as discussed in section 4.3

4.3 Combining the votes

The votes can be integrated using different ways to estimate the 3D representation Π^{m} of a 2D mono π^{m}:

- *Weighted averaging:*

$$\Pi^{\mathrm{m}} = C \sum_{i=1}^{N_P} v_i\, r_i, \tag{12}$$

  where C is a normalization constant.

- *Clustering:*
  Weighted averaging is prone to outliers which can be overcome by utilizing the set of clusters in the

votes. Let us denote the clusters by c_i for $i = 1, ...N_c$. Then, one integration scheme would be to take the cluster that has the highest average reliability:

$$\Pi^{\mathrm{m}} = \arg\max_{c_i} \ \frac{1}{\#c_i} \sum_{v_j \in c_i} r_j. \tag{13}$$

where r_i is the reliability (*i.e.*, confidence) associated to the vote v_i.

An alternative can use the most crowded cluster:

$$\Pi^{\mathrm{m}} = \arg\max_{c_i} \ \#c_i. \tag{14}$$

It is also possible to combine the number of votes and the average reliability of a cluster for making a decision.

As mentioned above, weighted averaging is prone to outliers but is fast. Clustering the votes can filter outliers whereas is slow. Moreover, clustering is an ill-posed problem, and most of the time, it is not trivial to determine the number of clusters from the data points that will be clustered.
In this paper, we implemented (1) a histogram-based clustering where the number of bins is fixed, and the best cluster is considered to be the bin with the most number of elements, and (2) a clustering algorithm where the number of clusters is determined automatically by making use of a cluster-regularity measure and maximizing this measure iteratively.
(1) is a simple but fast approach whereas (2) is considerably slower due to the iterative-clustering step. Suprisingly, our investigations showed that (1) and (2) produce almost identical results (the comparative results are not provided in this paper). For this reason, we have adopted (1) as the clustering method for the rest of the paper.

4.4 Combining the predictions using area information

3D surfaces project as areas into 2D images. Although one surface may project as many areas in the 2D image, it can be claimed that the image points in an image area are part of the same 3D surface[SK: This assumption does not always hold. I need to elaborate.].
Figure 10 shows the predictions of a surface. Due to strong outliers in the stereo computation, depth predictions are scattered around the surface that they are supposed to represent. We show that it is possible to segment the 2D image into areas based on intensity similarity and combine the predictions in areas to get a cleaner and more complete surface prediction.
We segment an input image $\boldsymbol{\mathcal{I}}$ into areas A_i, $i = 1, .., N_A$ using co-colority (see section 3) between primitives utilizing a simple region-growing method; the areas are grown until the image boundary or an edge-like primitive is hit. Figure 11 shows the segmentation of one of the images from figure 1.
In this paper, we assume that each A_i has a corresponding surface S_i defined as follows:

$$S_i(x, y, z) = \mathrm{a}x^2 + \mathrm{b}y^2 + \mathrm{c}z^2 + \mathrm{d}xy + \mathrm{e}yz + \mathrm{f}xz + \mathrm{g}x + \mathrm{h}y + \mathrm{i}z = 1. \tag{15}$$

Such a surface model allows a wide range of surfaces to be represented, including spherical, ellipsoid, quadratic, hyperbolic, conic, cylinderic and planar surfaces.
S_i is estimated from the predictions in A_i by solving for the coefficients using a least-squares method. As there are nine coefficients, such a method requires at least nine predictions to be available in area A_i. For the predictions shown in figure 10, the following surface is estimated which is shown in figure 12 using a sparse sampling (only non-zero coefficients are shown):

$$S_0 = 1.5 \times 10^{-5} y^2 + 5 \times 10^{-6} yz - 1.9 \times 10^{-4} x + 8 \times 10^{-3} y + 1.2 \times 10^{3} z = 1. \tag{16}$$

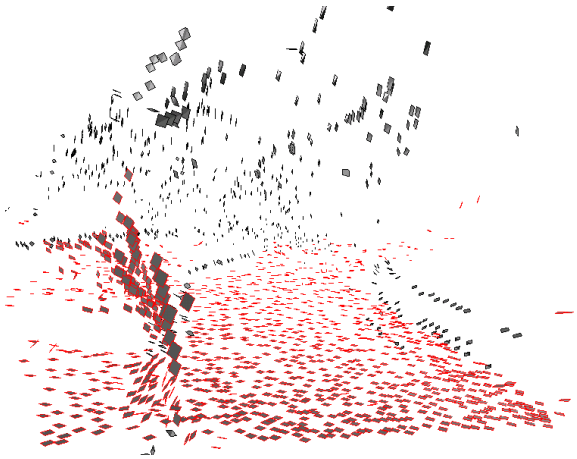


Figure 10: The predictions on the surface of the road for the input images shown in figure 1 (predictions are marked with red boundaries). The predictions are scattered around the plane of the road, and there are wrong predictions due to strong outliers in the computed stereo.

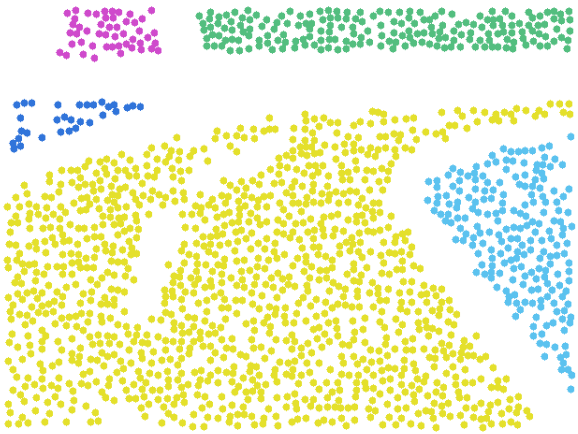


Figure 11: Segmentation of one of the input images given in 1 into areas using region-growing based on primitives.

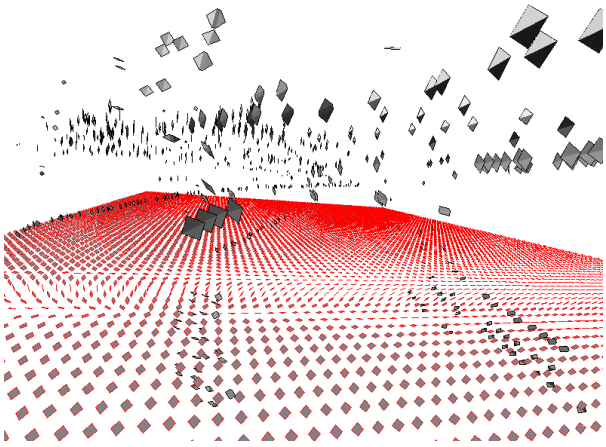


Figure 12: The surface given in equation 16 which is extracted from the predictions shown in figure 10.

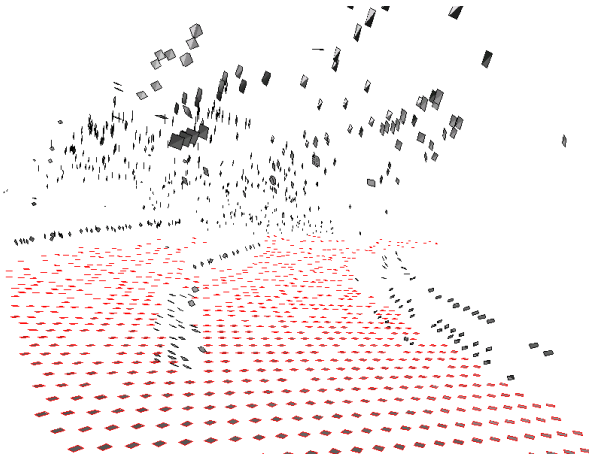


Figure 13: The predictions from 10 that are corrected using the extracted surface S_0 shown in equation 16 and figure 12.

S_0 in equation 16 is mainly a planar surface with small quadratic coefficients caused by outliers.
Having an estimated S_i for an area A_i, it is possible to *correct* the mono predictions using the estimated surface S_i: Let $\mathbf{X}_n$ be the intersection of the surface S_i with the ray that goes through π^{m} and the camera, and $\mathbf{n}_n$ be the surface normal at this point (defined by $\mathbf{n}_n = (\delta S_i/\delta_x, \delta S_i/\delta_y, \delta S_i/\delta_z)$). $\mathbf{X}_n$ and $\mathbf{n}_n$ are respectively the corrected position and the orientation of mono Π^{m}.
Corrected 3D monos for the example scene is shown in figure 13. Comparison with the initial predictions which are shown in figure 10 concludes that (1) outliers are *corrected* with the extracted surface representation, and (2) orientations and positions are qualitatively better.

5 Results

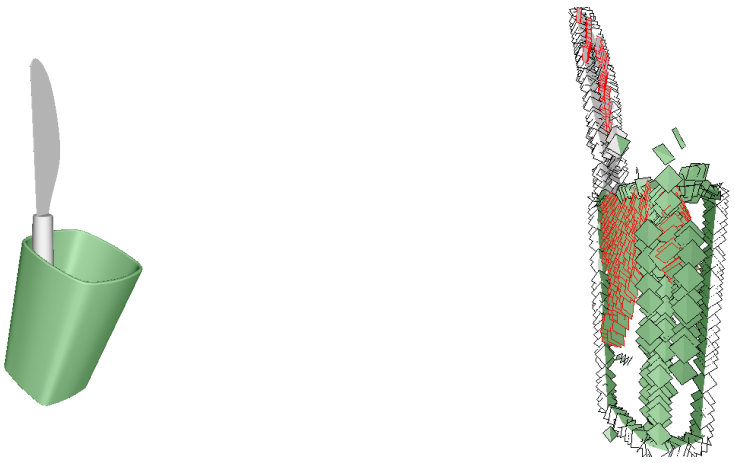
The test cases include kitchen scenarios and road scenarios which are intended for PACO+ and Drivsco projects, respectively. The results of our model is shown for a few examples in figures 14, 15, 16 and 17.
The results show that inspite of limited 3D information from feature-based stereo which may contain strong outliers in some of the scenes (as shown in figure 1), our result is able to predict the surfaces.

6 Conclusion

In this paper, we introduced a voting model that estimates the depth at homogeneous or weakly-textured image patches (called monos) from the depth of the bounding edge-like structures. The depth at edge-like structures is computed using a feature-based stereo algorithm [20], and is used to vote for the depth of a mono, which otherwise is not possible to compute easily due to the correspondence problem.
The method presented in this paper is an ongoing work. In the future, the reliability of each vote will be replaced by the statistics collected from chromatic range data (see [12]). Moreover, comprehensive comparison as well as possible combination with dense stereo methods are going to be investigated.

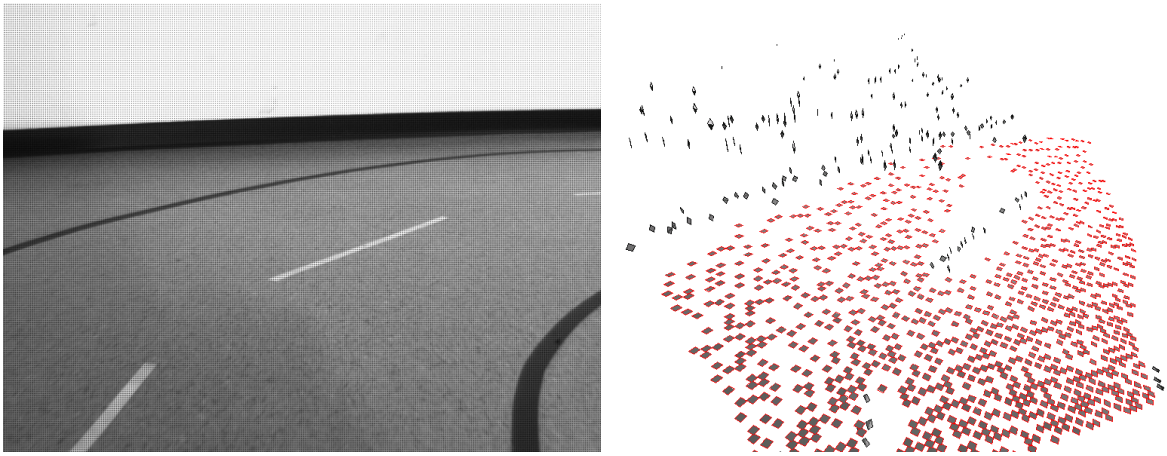
7 Acknowledgments

This work is supported by Drivsco projects.



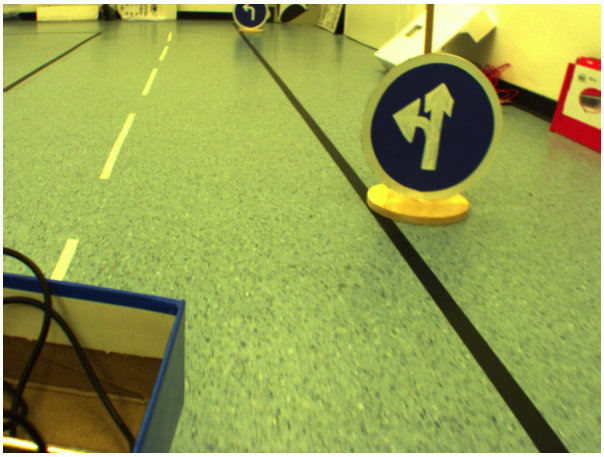
(a) (b)

Figure 14: Experiment results on an artificial *kitchen* scene. **(a)** Left image of the input stereo pair. **(b)** The predictions of our model.



(a) (b)

Figure 15: Experiment results on a road scene. **(a)** Left image of the input stereo pair. **(b)** The predictions of our model.



(a)

(b)

Figure 16: Experiment results on a road scene. **(a)** Left image of the input stereo pair. **(b)** The predictions of our model.

(a)

(b)

Figure 17: Experiment results on a *kitchen* scene. **(a)** Left image of the input stereo pair. **(b)** The predictions of our model.

References

[1] H. G. Barrow and J. M. Tenenbaum. Interpreting line drawings as three-dimensional surfaces. *Artificial Intelligence*, 17:75–116, 1981.

[2] A. B.L., S. M., and F. R.W. The interpolation of object and surface structure. *Cognitive Psychology*, 44:148–190(43), March 2002.

[3] V. Bruce, P. R. Green, and M. A. Georgeson. *Visual Perception: Physiology, Psychology and Ecology*. Psychology Press, 4th edition, 2003.

[4] T. S. Collett. Extrapolating and Interpolating Surfaces in Depth. *Royal Society of London Proceedings Series B*, 224:43–56, Mar. 1985.

[5] O. Faugeras. *Three–Dimensional Computer Vision*. MIT Press, 1993.

[6] W. E. L. Grimson. A Computational Theory of Visual Surface Interpolation. *Royal Society of London Philosophical Transactions Series B*, 298:395–427, Sept. 1982.

[7] W. E. L. Grimson. Binocular shading and visual surface reconstruction. *Computer Vision, Graphics, and Image Processing*, 28(1):19–43, 1984.

[8] G. Guy and G. Medioni. Inference of surfaces from sparse 3-d points. In *ARPA94*, pages II:1487–1494, 1994.

[9] D. Hubel and T. Wiesel. Anatomical demonstration of columns in the monkey striate cortex. *Nature*, 221:747–750, 1969.

[10] S. Kalkan, N. Pugeault, M. Christiansen, and N. Krüger. Relations between primitives. Technical report, University of Southern Denmark, 2006. (to be submitted).

[11] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of local 3d structure in 2d images. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1114–1121, 2006.

[12] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of second-order relations of 3d structures. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[13] N. Krüger and M. Felsberg. An explicit and compact coding of geometric and structural information applied to stereo matching. *Pattern Recognition Letters*, 25(8):849–863, 2004.

[14] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 1(5):417–428, 2004.

[15] N. Krüger, N. Pugeault, and F. Wörgötter. Multi-modal primitives: Local, condensed, and semantically rich visual descriptors and the formalization of contextual information. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-4, 2007.

[16] N. Krüger, N. Pugeault, and F. Wörgötter. Multi-modal primitives: Local, condensed, and semantically rich visual descriptors and the formalization of contextual information. *To be submitted.*, 2007.

[17] M. S. Lee and G. Medioni. Inferring segmented surface description from stereo data. In *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 346, Washington, DC, USA, 1998. IEEE Computer Society.

[18] M.-S. Lee, G. Medioni, and P. Mordohai. Inference of segmented overlapping surfaces from binocular stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(6):824–837, 2002.

[19] V. S. Nalwa. Line-drawing interpretation: Bilateral symmetry. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(10):1117–1120, 1989.

[20] N. Pugeault and N. Krüger. Multi–modal matching applied to stereo. *Proceedings of the BMVC 2003*, pages 271–280, 2003.

[21] N. Pugeault, F. Wörgötter, and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proc. IEEE Workshop on Perceptual Organization in Computer Vision (in conjunction with CVPR'06)*, 2006.

[22] K. A. Stevens. The visual interpretations of surface contours. *Artificial Intelligence*, 17:47–73, 1981.

[23] D. Terzopoulos. Multi-level reconstruction of visual surfaces: Variational principles and finite element representations. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1982.

[24] D. Terzopoulos. The computation of visible-surface representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(4):417–438, 1988.

[25] F. Ulupinar and R. Nevatia. Constraints for interpretation of line drawings under perspective projection. *CVGIP: Image Underst.*, 53(1):88–96, 1991.

[26] F. Ulupinar and R. Nevatia. Perception of 3-d surfaces from 2-d contours. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(1):3–18, 1993.

STATISTICAL ANALYSIS OF SECOND-ORDER RELATIONS OF 3D STRUCTURES

Published at VISAPP'07.

Preparation of Camera-Ready Contributions to INSTICC Proceedings

Sinan Kalkan, Florentin Wörgötter

Bernstein Centre for Computational Neuroscience, Unv. of Göttingen, Germany

{*sinan,worgott*}*@bccn-goettingen.de*

Norbert Krüger

Cognitive Vision Group, Unv. of Southern Denmark, Denmark

norbert@mip.sdu.dk

Keywords: Range Data Statistics, Indirect Depth Estimation

Abstract: Algorithmic 3D reconstruction methods like stereopsis or structure from motion fail to extract depth at homogeneous image structures where the human visual system succeeds and is able to estimate depth. In this paper, using chromatic 3D range data, we analyze in which way depth in homogeneous structures is related to the depth at the bounding edges. For this, we first extract the local 3D structure of regularly sampled points, and then, analyze the coplanarity relation between these local 3D structures. We can statistically show that the likelihood to find a certain depth at a homogeneous image patch depends on the distance between the image patch and its edges. Furthermore, we find that this prediction is higher when there is a second edge which is proximate to and coplanar with the first edge. These results allow deriving statistically based prediction models for depth extrapolation into homogeneous image structures. We present initial results of a model that predicts depth based on these statistics.

1 INTRODUCTION

Depth estimation relies on the extraction of 3D structure from 2D images which is realized by a set of inverse problems including structure from motion, stereo vision, shape from shading, linear perspective, texture gradients and occlusion (Bruce et al., 2003). In methods which make use of multiple views (*i.e.*, stereo and structure from motion), correspondences between different 2D views of the scene are required. In contrast, monocular or pictorial cues such as shape from shading, utilization of texture gradients or linear perspective use statistical and geometrical relations in one image to make statements about the underlying 3D structure.

Many surfaces have only weak texture or no texture at all, and as a consequence, the correspondence problem is very hard or not at all resolvable for these surfaces. Nevertheless, humans are able to reconstruct 3D information for these surfaces, too. This gives rise to the assumption that in the human visual system, an interpolation process is realized that starting with the local analysis of edges, corners and textures, computes depth also in areas where correspondences cannot easily be found.

In figure 1, the relation between the depth of homogeneous image structures and edges is shown. In figure 1(a), we see that the depth of homogeneous image structures is directly related to the depth of the bounding edges; however, this relation does not always exist as shown in figure 1(b,c) where the depth is cued in shading.

With the notion that the human visual system is adapted to the statistics of the environment (Brunswik and Kamiya, 1953; Knill and Richards, 1996; Krüger, 1998; Krüger and Wörgötter, 2004; Olshausen and Field, 1996; Rao et al., 2002; Purves and Lotto, 2002) and its successful applications to grouping, object recognition and stereo (Elder and Goldberg, 2002; Elder et al., 2003; Pugeault et al., 2004; Zhu, 1999), the analysis, and the usage of natural image statistics has become an important focus of vision research. Moreover, with the advances in technology, it has been also possible to analyze the underlying 3D world using 3D range scanners (Howe and Purves, 2004; Huang et al., 2000; Potetz and Lee, 2003; Yang and Purves, 2003).

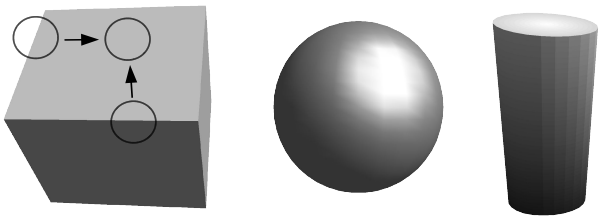
In this paper, by making use of chromatic range data (see figure 3 for examples), we investigate

whether the depth at homogeneous image structures are related to or predictable by the depth of the edges that bound them. This investigation is important because (1) it contributes to a better understanding of the intrinsic parameters of the 3D world, and (2) it suggests an indirect method to estimate the depth for homogeneous image structures; that is, using the depth estimations about the edges to predict the depth of homogeneous image structures instead of using the 2D image information itself as shown in figure 1(a).

There have been only a few studies that have investigated the 3D world from range data (Howe and Purves, 2004; Huang et al., 2000; Kalkan et al., 2006; Potetz and Lee, 2003; Yang and Purves, 2003). In (Yang and Purves, 2003), the distribution of roughness, size, distance, 3D orientation, curvature and independent components of surfaces was analyzed. Their major conclusions were: (1) local 3D patches tend to be saddle-like, and (2) natural scene geometry is quite regular and less complex than luminance images. In (Huang et al., 2000), the distribution of 3D points was analyzed using co-occurrence statistics and 2D and 3D joint distributions of Haar filter reactions. They showed that range images are much simpler to analyze than optical images and that a 3D scene is composed of piecewise smooth regions. In (Potetz and Lee, 2003), the correlation between light intensities of the image data and the corresponding range data as well as surface convexity were investigated. They could justify the event that brighter objects are closer to the viewer, which is used in shape from shading algorithms for estimating depth. In (Howe and Purves, 2002; Howe and Purves, 2004), range image statistics were analyzed for explanation of several visual illusions.

In (Kalkan et al., 2006), a higher-order representation of the 2D local image patches and the 3D local patches were considered; they represented 2D images in terms of homogeneous, edge-like and corner-like structures whereas 3D range data in terms of continuities, gap discontinuities and orientation discontinuities (see section 2). With these representations, they could compute the probability $P(\text{3D Structure} \mid \text{2D Structure})$ which among other things justifies and quantifies the assumption that if two image points do not have contrast difference in-between, then they are likely to be coplanar. This assumption is called 'no news is good news' and widely used in 3D reconstruction studies (see, *e.g.*, (Grimson, 1983)).

All the studies discussed above are first-order, analyzing the relation between the image data and the range data. In this work, however, we are interested in higher order relations between local 3D features. In this sense, our work is a natural extension of (Kalkan et al., 2006).



(a) (b) (c)

Figure 1: Illustration of the relation between the depth of homogeneous image structures and the bounding edges. (a) In the case of cube, the relation is eminent. However, in the case of round surfaces, (b) the depth of homogeneous image structures may not be related to the depth of the bounding edges. (c) In the case of a cylinder, we see both cases of the relation as illustrated in (a) and (b).

The outline of the paper is as follows: In section 2, different types of local 3D structures are introduced. In section 3, the methodology underlying our statistical analysis is presented. The results are presented and discussed in section 4. Finally, in section 5, the paper is concluded.

2 LOCAL 3D STRUCTURE TYPES

For our work, we have made use of the classification introduced in (Kalkan et al., 2006) where it is intuitively argued that the local 3D structure of a point can be:

- Surface Continuity: The underlying 3D structure can be described by one surface whose normal does not change or changes smoothly.
- Regular Gap discontinuity: The underlying 3D structure can be described by a small set of surfaces with a significant depth difference. An example of gap discontinuity is shown in figure 2(d).
- Irregular Gap discontinuity: The underlying 3D structure shows high depth variation and cannot be described by two or three surfaces. An example of an irregular gap discontinuity is shown in figure 2(e).
- Orientation Discontinuity: The underlying 3D structure can be described by two surfaces with significantly different 3D orientations that meet at the point whose 3D structure is being questioned. In this type of discontinuity, no gap but a change

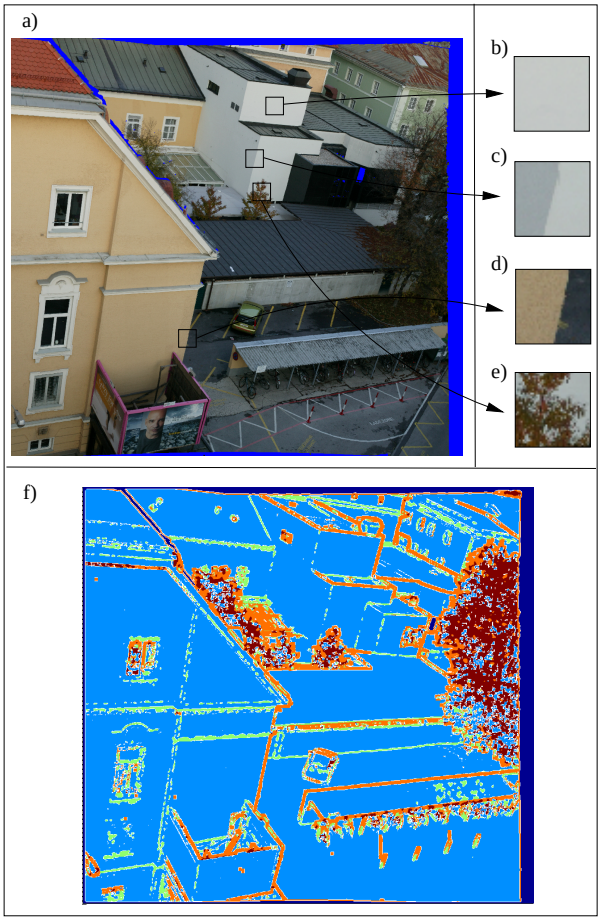


Figure 2: Illustration of the types of 3D discontinuities. (a) 2D image. (b) Continuity. (c) Orientation discontinuity. (d) Gap discontinuity. (e) Irregular gap discontinuity. (f) 3D discontinuity of each pixel is shown in different colors. Blue: continuous surfaces, light blue: orientation discontinuities, orange: gap discontinuities and brown: irregular gap discontinuities. Dark blue indicates points without range data.

in 3D orientation between the meeting surfaces occurs. An example for this type of discontinuity is shown in figure 2(c).

3D discontinuities are detected in studies which involve range data processing, using different methods and using different names like two-dimensional discontinuous edge, jump edge or depth discontinuity for gap discontinuity; and, two-dimensional corner edge, crease edge or surface discontinuity for orientation discontinuity (Bolle and Vemuri, 1991; Hoover et al., 1996; Shirai, 1987).

For our analysis, we have adopted the measures defined in (Kalkan et al., 2006). In this work, a gap discontinuity is measured by simple edge detection in XYZ coordinate values. An orientation discontinuity is measured by exploiting the fact that two meeting surfaces with different orientations produce two clusters in the histogram distribution of the 3D orientation of the points. An irregular discontinuity is measured by exploiting the fact that the histogram distribution of the 3D orientation of the points should be flat.

Discontinuity types of each pixel for a scene is shown in figure 2(f) where the local 3D structure type of each point is shown in different colors.

Figure 3: A subset of the 20 3D data sets used in the analysis. The points without corresponding range data are marked in blue. The gray image shows the range data of the top-left scene. The resolution range of the whole data set is [512-2048]x[390-2290] with an average resolution of 1140x1001.

3 METHODS

In our analysis, we used chromatic range data of outdoor scenes which were obtained from Riegl UK Ltd. (`http://www.riegl.co.uk/`). There were 20 scenes in total; due to space limitations, only two of them are shown in figure 3. The range of an object which does not reflect the laser beam back to the scanner or which is out of the range of the scanner cannot be measured. These points are marked with blue in figure 3 and are not processed in our analysis. The resolution range of the data set is [512-2048]x[390-2290] with an average resolution of 1140x1001.

3.1 Representation

Using the 2D image and the associated 3D range data, a representation of the scene is created in terms of local compository 2D and 3D features denoted by π. For homogeneous and edge-like structures, different representations are needed due to different underlying structures (in the rest of the paper, a homogeneous image structure that corresponds to a 3D continuity will be called a *mono*.). For this reason, we have two different definitions of π denoted respectively by π^E (for edge-like structures) and π^M (for monos) and formulated as:

$$\pi^M = (\mathbf{X}_{3D}, \mathbf{X}_{2D}, \mathbf{c}, \mathbf{p}), \quad (1)$$

$$\pi^E = (\mathbf{X}_{3D}, \mathbf{X}_{2D}, \phi_{2D}, \mathbf{c}_1, \mathbf{c}_2, \mathbf{p}_1, \mathbf{p}_2), \quad (2)$$

where $\mathbf{X}_{3D}$ and $\mathbf{X}_{2D}$ denote 3D and 2D positions of the 3D entity; ϕ_{2D} is 2D orientation of the 3D entity; $\mathbf{c}_1$ and $\mathbf{c}_2$ are the 2D color representation of the surfaces of the 3D entity; $\mathbf{c}$ represents the color of π^M; $\mathbf{p}_1$ and $\mathbf{p}_2$ are the planes that represent the surfaces that meet at the 3D entity; and $\mathbf{p}$ represents the plane of π^M (see figure 4). Note that π^M does not have any 2D orientation information (because it is undefined for homogeneous structures), and π^E has two color and plane representations to the 'left' and 'right' of the edge.

The process of creating the representation of a scene is illustrated in figure 4.

In our analysis, the entities are regularly sampled from the 2D information. The sampling size is 10 pixels. See (Krüger et al., 2003; Krüger and Wörgötter, 2005) for details.

Extraction of the planar representation requires knowledge about the type of local 3D structure of the 3D entity (see figure 4). Namely, if the 3D entity is a continuous surface, then only one plane needs to be extracted; if the 3D entity is an orientation discontinuity, then there will be two planes for extraction; if the 3D entity is a gap discontinuity, then there will also be two planes for extraction.

In the case of a continuous surface, a single plane is fitted to the set of 3D points in the 3D entity in question. For orientation discontinuous 3D structures, extraction of the planar representation is not straightforward. For these structures, our approach was to fit unit-planes[1] to the 3D points of the 3D entity and find the two clusters in these planes using k-means clustering of the 3D orientations of the small planes. Then, one plane is fitted for each of the two clusters, producing the two-fold planar representation of the 3D entity.

[1]By unit-planes, we mean planes that are fitted to the 3D points that are 1-pixel apart in the 2D image.

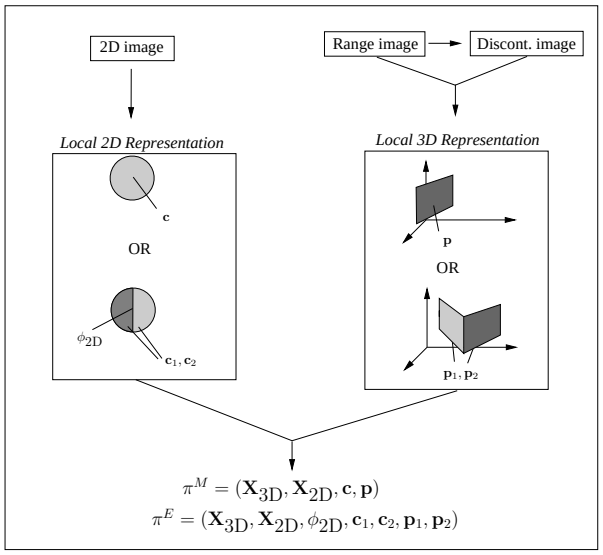


Figure 4: Illustration of the representation of a 3D entity. From the 2D and 3D information, local 2D and 3D representation is extracted.

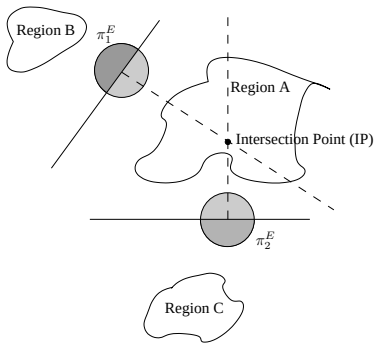


Figure 5: The parameters involved in second order 3D statistics.

Color representation is extracted in a similar way. If the image patch is a homogeneous structure, then the average color of the pixels in the patch is taken to be the color representation. If the image patch is edge-like, then it has two colors separated by the line which goes through the center of the image patch and which has the 2D orientation of the image patch. In this case, the averages of the colors of the different sides of the edge define the color representation in terms of c_1 and c_2. If the image patch is corner-like, the color representation becomes undefined.

3.2 Collecting the Data Set

In our analysis, we form pairs out of π^Es that are close enough, and for each pair, we check whether monos in the scene are coplanar to the elements of the pair

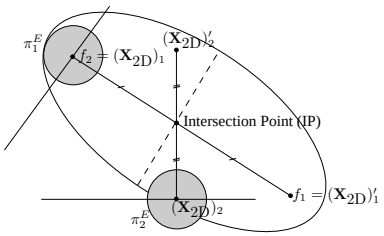


Figure 6: The ellipse in second order 3D statistics.

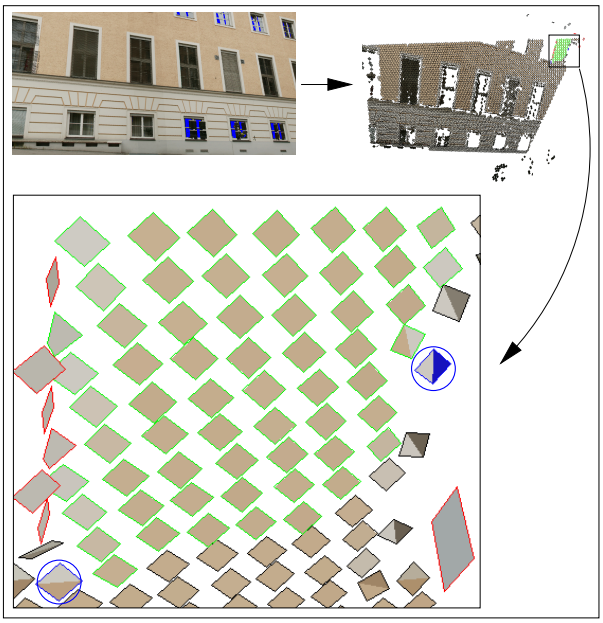


Figure 7: Illustration of a pair of π^E and the set of monos associated to them. Top-left shows the 2D image. Top-right shows the 3D representation of the scene in our 3D visualization software. At the bottom, a part of the 3D representation is displayed in detail where the edges are shown in blue; the monos coplanar with the edges are shown in green, and non-coplanar monos are shown in red. The entities are drawn in rectangles because of the high computational complexity of drawing circles.

or not. As there are plenty of monos in the scene, we only consider a subset of monos for each pair of π^E that we suspect to be relevant to the analysis because otherwise, the analysis becomes computationally intractable. The situation is illustrated in figure 5. In this figure, two π^E and three regions are shown; however, only one of these regions (*i.e.*, region A) is likely to have coplanar monos (*e.g.*, see figure 1(a)).

Let $\mathcal{P}$ denote the set of pairs of proximate π^Es whose normals intersect. $\mathcal{P}$ can be defined as:

$$\mathcal{P} = \left\{(\pi_1^E, \pi_2^E) \mid \forall \pi_1^E, \pi_2^E,\ \pi_1^E \in \Omega(\pi_2^E),\ I(\perp(\pi_1^E), \perp(\pi_2^E))\right\}, \tag{3}$$

where $\Omega(\pi^E)$ is the N-pixel-2D-neighborhood of π^E; $\perp(\pi^E)$ is the 2D line orthogonal to the 2D orientation of π^E, i.e., the normal of π^E; and, $I(l_1, l_2)$ is true if the lines l_1 and l_2 intersect. We have taken N to be 100.

Next, we have to determine which monos in region A should be analyzed for the relation; that is, what is the shape of region A? Empirically, it turns out that an ellipse (1) is the computationally cheapest shape and (2) fits to different configurations of π_1 and π_2 under different orientations and distances. Neither a rectangle nor a circle satisfy these two properties. Figure 6 demonstrates the ellipse for the example pair of edges in figure 5. The center of the ellipse is at the intersection of the normals of the edges which we call as the intersection point (IP) in the rest of the paper.

For each pair of edges in $\mathcal{P}$, we decide on which region to analyze the relation of depth by intersecting the normals of the edges. Then, we associate the monos inside the ellipse that are defined by the pair of edges.

Note that a π^E has two planes that represent the underlying 3D structure. When π^Es become associated to monos, only one plane that faces the ellipse becomes relevant. Let π^{sE} denote the semi-representation of π^E which can be defined as:

$$\pi^{sE} = (\mathbf{X}_{3D}, \mathbf{X}_{2D}, \mathbf{c}, \mathbf{p}). \tag{4}$$

Note that π^{sE} is equivalent to the definition of π^M in equation 2.

Let $\mathcal{T}$ denote the data set which stores $\mathcal{P}$ and the associated monos which can be formulated as:

$$\mathcal{T} = \{(\pi_1^{sE}, \pi_2^{sE}, \pi^M) \mid (\pi_1^E, \pi_2^E) \in \mathcal{P}, \pi^M \in \mathcal{S}^M, \pi^M \in E(\pi_1^E, \pi_2^E)\}, \tag{5}$$

where $\mathcal{S}^M$ is the set of all π^M, and $E(\pi_1^E, \pi_2^E)$ represents the ellipse associated to π_1^E and π_2^E[2].

A pair of π^Es and the set of monos associated to them are illustrated in figure 7. The edges are shown in blue, and the coplanar and non-coplanar monos are shown in green and red, respectively.

[2]The parameters of an ellipse are composed of two focus points f_1, f_2 and the minor axis b. In our analysis, the more distant 3D edge determines the foci of the ellipse (and, hence, the major axis), and the other 3D edge determines the minor axis.

Let us denote the position of two 3D edges π_1^E, π_2^E by $(\mathbf{X}_{2D})_1$ and $(\mathbf{X}_{2D})_2$ respectively. The vectors between the 3D edges and IP (let us call l_1 and l_2) can be defined as:

$$\begin{aligned} l_1 &= ((\mathbf{X}_{2D})_1 - IP), \\ l_2 &= ((\mathbf{X}_{2D})_2 - IP). \end{aligned} \tag{6}$$

Having defined l_1 and l_2, the ellipse $E(\pi_1^E, \pi_2^E)$ is as follows:

$$E(\pi_1^E, \pi_2^E) = \begin{cases} f_1 = (\mathbf{X}_{2D})_1, f_2 = (\mathbf{X}_{2D})'_1, b = |l_2| & \text{if } |l_1| > |l_2|, \\ f_1 = (\mathbf{X}_{2D})_2, f_2 = (\mathbf{X}_{2D})'_2, b = |l_1| & \text{otherwise.} \end{cases} \tag{7}$$

where $(\mathbf{X}_{2D})'$ is the symmetry of $\mathbf{X}_{2D}$ around the intersection point and on the line defined by $\mathbf{X}_{2D}$ and IP (as shown in figure 6).

3.3 Definition of Coplanarity

Let π^s denote either a semi-edge π^{sE} or a mono π^M. Two π^s are coplanar iff they are on the same plane. When it comes to measuring coplanarity, two criteria need to be applied:

$$cop(\pi_1^s,\pi_2^s) = \alpha(\mathbf{p}^{\pi_1^s},\mathbf{p}^{\pi_2^s}) < T_p \text{ AND } d(\mathbf{p}^{\pi_1^s},\pi_2^s)/d(\pi_1^s,\pi_2^s) < T_d, \quad (8)$$

where $\mathbf{p}^{\pi^s}$ is the plane associated to π^s; $\alpha(\mathbf{p}_1,\mathbf{p}_2)$ is the angle between the orientations of $\mathbf{p}_1$ and $\mathbf{p}_2$; and, $d(.,.)$ is the Euclidean distance between two entities.

In our analysis, we have empirically chosen T_p and T_d as 20 and 0.5, respectively.

4 RESULTS AND DISCUSSIONS

The data set consists of pairs of π_1^E, π_2^E and the associated monos. Using this set, we compute the likelihood that a mono is coplanar with π_1^E and/or π_2^E against a distance measure.

Figure 8 shows the results of our analysis. In figure 8(a), the likelihood of the coplanarity of a mono against the distance to π_1^E *or* π_2^E is shown. This likelihood can be denoted formally as $P(cop(\pi^M,\pi_1^E \wedge \pi_2^E) \mid d_N(\pi^M,\pi^E))$ where $cop(\pi^M,\pi_1^E \wedge \pi_2^E)$ is defined as $cop(\pi_1^E,\pi_2^E) \wedge cop(\pi^M,\pi^E)$, and π^E is either π_1^E or π_2^E. The normalized distance measure[3] $d_N(\pi^M,\pi^E)$ is defined as:

$$d_N(\pi^M,\pi^E) = \frac{d(\pi^M,\pi^E)}{2\sqrt{d(\pi_1^E,IP)^2 + d(\pi_2^E,IP)^2}}, \quad (9)$$

where π^E is either π_1^E or π_2^E, and IP is the intersection point of π_1^E and π_2^E. We see in figure 8(a) that the likelihood decreases when a mono is more distant from an edge. However, when the distance measure gets closer to 1, the likelihood increases. This is because when the mono gets away from either π_1^E or π_2^E, it becomes closer to the other π^E.

In figure 8(b), we see the unconstrained case of figure 8(a); *i.e.,* the case where there is no information about the coplanarity of π_1^E and π_2^E, namely, $P(cop(\pi^M,\pi^E) \mid d_N(\pi^M,\pi^E))$ where π^E is either π_1^E or π_2^E. We see that the likelihood distribution is weaker than the case where π_1^E and π_2^E are coplanar. The comparison with figure 8(a) shows that the existence of another edge in the neighborhood increases the likelihood of finding coplanar structures.

[3]In the following plots, the distance means the Euclidean distance in the image domain.

In figure 8(c), the likelihood of the coplanarity of a mono against the distance to IP (*i.e.,* $P(cop(\pi^M,\pi_1^E \wedge \pi_2^E) \mid d_{NU}(\pi^M,IP), d_{NV}(\pi^M,IP))$) is shown. We see in the figure that the likelihood shows a flat distribution against the distance to IP.

In figure 8(d), the likelihood of the coplanarity of a mono against the distance to π_1^E *and* π_2^E (*i.e.,* $P(cop(\pi^M,\pi_1^E \wedge \pi_2^E) \mid d_N(\pi^M,\pi_1^E), d_N(\pi^M,\pi_2^E))$) is shown. We see that when π^M is close to π_1^E or π_2^E, it is more likely to be coplanar with π_1^E and π_2^E than when it is equidistant to both edges. The reason is when π^M moves away from an equidistant point, it becomes closer to the other edge and in that case, as shown in figure 8(a), the likelihood increases.

The results, especially figure 8(a) and (b) confirm the importance of the relation illustrated in figure 1(a).

In figure 9, first results of an unpublished ongoing work on a depth prediction model based on the presented statistical framework are presented. 9(c) shows the results of feature-based stereo while in 9(d), depth predictions are shown in our 3D display software

5 CONCLUSION

In this paper, using 3D range data with real-world color information, we have analyzed whether the depth of a mono is predictable from the depth of the edges that bound the homogeneous image patch. We have analyzed the predictability of the depth of a mono given a single edge and a pair of coplanar edges.

We have shown that a mono is more likely to be coplanar with an edge when it is closer to the edge and when there is another coplanar edge in the neighborhood. We have shown that the existence of a coplanar edge in the neighborhood is a strong event and to our knowledge, is not recognized by the literature.

The results suggest that the depth estimation at homogeneous image structures can be achieved indirectly from the available information at the edges. We believe that this is a new approach to 3D reconstruction.

In this paper, we are only interested in second-order long-range relations between local features. For round objects like shown in figure 1(b,c), the depth information is given by the shading whose statistical properties can only be captured by different relations.

In our current work, we are developing a model that exploits the statistics presented in this paper to predict the depth of homogeneous image patches from the depth of edges. First results of this ongoing work are also presented in the paper.

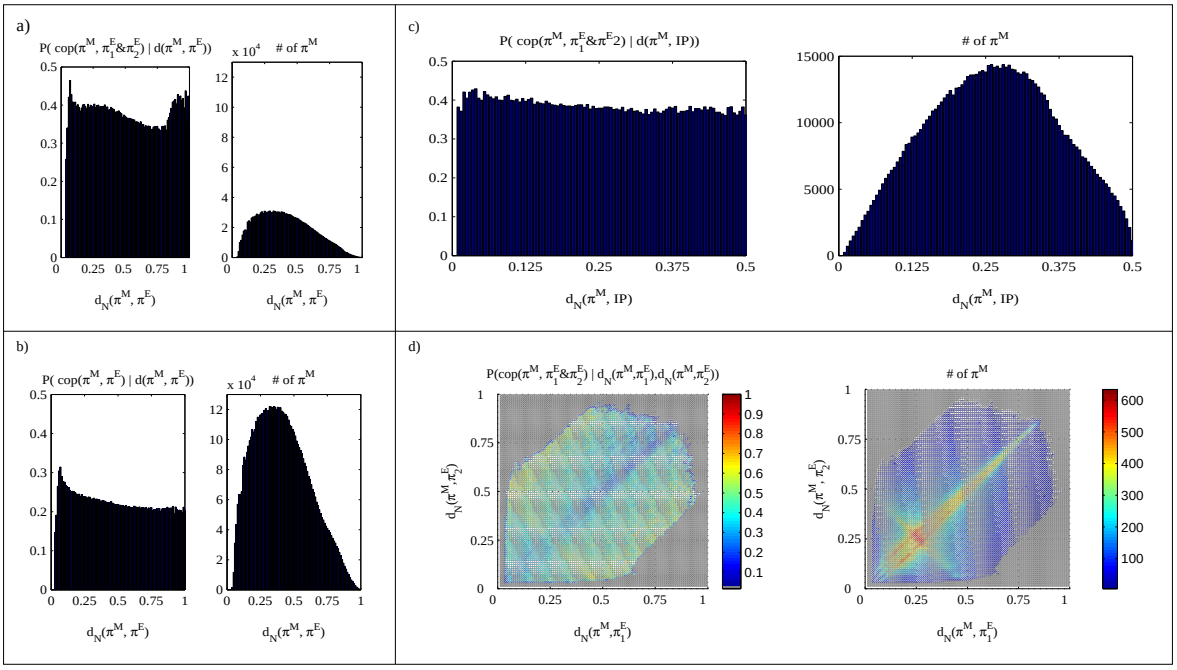


Figure 8: Likelihood distribution of coplanarity of monos. In each sub-figure, left-plot shows the likelihood distribution whereas right-plot shows the frequency distribution. (a) The likelihood of the coplanarity of a mono with π_1^E *and* π_2^E against the distance to π_1^E *or* π_2^E.(b) The likelihood of the coplanarity of a mono with π_1^E *or* π_2^E against the distance to π_1^E *or* π_2^E.(c) The likelihood of the coplanarity of a mono against the distance to *IP*. (d) The likelihood of the coplanarity of a mono against the distance to π_1^E *and* π_2^E.

6 ACKNOWLEDGEMENTS

We would like to thank RIEGL UK Ltd. for providing us with 3D range data. This work is supported by the Drivsco and NISA projects.

REFERENCES

Bolle, R. M. and Vemuri, B. C. (1991). On three-dimensional surface reconstruction methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1):1–13.

Bruce, V., Green, P. R., and Georgeson, M. A. (2003). *Visual Perception: Physiology, Psychology and Ecology*. Psychology Press, 4th edition.

Brunswik, E. and Kamiya, J. (1953). Ecological cue–validity of 'proximity' and of other Gestalt factors. *American Journal of Psychologie*, LXVI:20–32.

Elder, H. and Goldberg, R. (2002). Ecological statistics of gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2(4):324–353.

Elder, J. H., Krupnik, A., and Johnston, L. A. (2003). Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(25):1–14.

Grimson, W. E. L. (1983). Surface consistency constraints in vision. *Computer Vision, Graphics and Image Processing*, 24(1):28–51.

Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P. J., Bunke, H., Goldgof, D. B., Bowyer, K., Eggert, D. W., Fitzgibbon, A., and Fisher, R. B. (1996). An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689.

Howe, C. Q. and Purves, D. (2002). Range image statistics can explain the anomalous perception of length. *PNAS*, 99(20):13184–13188.

Howe, C. Q. and Purves, D. (2004). Size contrast and assimilation explained by the statistics of natural scene geometry. *Journal of Cognitive Neuroscience*, 16(1):90–102.

Huang, J., Lee, A. B., and Mumford, D. (2000). Statistics of range images. *CVPR*, 1(1):1324–1331.

Kalkan, S., Wörgötter, F., and Krüger, N. (2006). Statistical analysis of local 3d structure in 2d images. *CVPR*, 1:1114–1121.

Knill, D. C. and Richards, W., editors (1996). *Perception as bayesian inference*. Cambridge: Cambridge University Press.

Krüger, N. (1998). Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129.

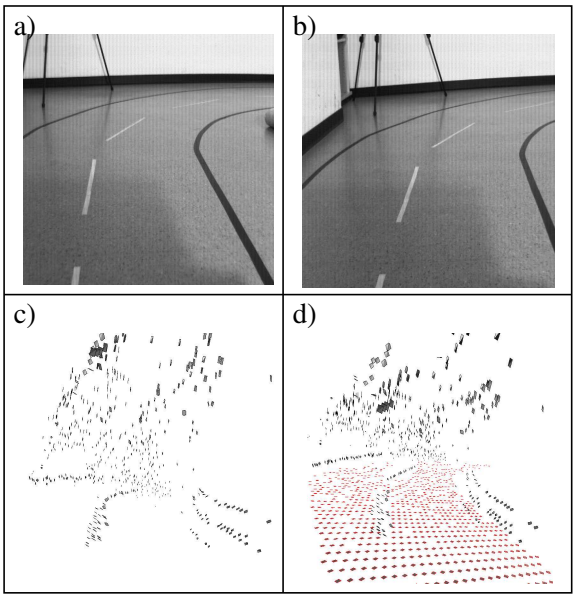


Figure 9: Initial depth prediction results on a toy example. (a) Left image. (b) Right image. (c) Feature-based stereo result giving depth only at edges. (d) Depth-prediction based on (c).

Krüger, N., Lappe, M., and Wörgötter, F. (2003). Biologically motivated multi-modal processing of visual primitives. *Proc. the AISB 2003 Symposium on Biologically inspired Machine Vision, Theory and Application, Wales*, pages 53–59.

Krüger, N. and Wörgötter, F. (2004). Statistical and deterministic regularities: Utilisation of motion and grouping in biological and artificial visual systems. *Advances in Imaging and Electron Physics*, 131:82–147.

Krüger, N. and Wörgötter, F. (2005). Multi-modal primitives as functional models of hyper-columns and their use for contextual integration. *Proc. 1st Int. Symposium on Brain, Vision and Artificial Intelligence, Naples, Italy, Lecture Notes in Computer Science, Springer, LNCS 3704*, pages 157–166.

Olshausen, B. and Field, D. (1996). Natural image statistics and efficient coding. *Network*, 7:333–339.

Potetz, B. and Lee, T. S. (2003). Statistical correlations between two-dimensional images and three-dimensional structures in natural scenes. *Journal of the Optical Society of America*, 20(7):1292–1303.

Pugeault, N., Krüger, N., and Wörgötter, F. (2004). A non-local stereo similarity based on collinear groups. *Proceedings of the Fourth International ICSC Symposium on Engineering of Intelligent Systems*.

Purves, D. and Lotto, B., editors (2002). *Why we see what we do: an empirical theory of vision*. Sunderland, MA: Sinauer Associates.

Rao, R. P. N., Olshausen, B. A., and Lewicki, M. S., editors (2002). *Probabilistic models of the brain*. MA: MIT Press.

Shirai, Y. (1987). *Three-dimensional computer vision*. Springer-Verlag New York, Inc.

Yang, Z. and Purves, D. (2003). Image/source statistics of surfaces in natural scenes. *Network: Computation in Neural Systems*, 14:371–390.

Zhu, S. C. (1999). Embedding gestalt laws in markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1170–1187.

Robotics Group
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Technical Report no. 2007 – 5

Using Tactile Sensors for Multisensorial Scene Exploration

Morten Kjærgaard, Alex Bierbaum, Dirk Kraft, Sinan Kalkan, Norbert Krüger, Tamim Asfour, Rüdiger Dillmann

January 22, 2007

Title	Using Tactile Sensors for Multisensorial Scene Exploration
Author(s)	Morten Kjærgaard, Alex Bierbaum, Dirk Kraft, Sinan Kalkan, Norbert Krüger, Tamim Asfour, Rüdiger Dillmann
Publication History	

1 Introduction

This paper explores the usage of two relatively cheap industrial haptic sensors to extend visual scene exploration to a multisensorial scene exploration. It takes a look at the extraction of different object properties and how these can be used to augment the existing visual representation of the environment.
Section 2 gives an overview of the requirements for a sensor for haptic exploration and the electrical characteristics of the considered sensors and the measurement electronics. In section 3 the physical properties of the sensor are shown and its step response and a resulting linearization are derived. Section 4 presents different experiments and approaches towards extracting surface normals, weight information and eleasticity from the sensor data. The fusion of visual predicted surface information with haptical surface exploration is shown in section 5.

2 Hardware

2.1 Choice of the Tactile Sensor Type

The purpose of the tactile sensor system is the support of haptic exploration and controlled grasping skills for the robot. The following requirements arise from this application:

High sensitivity and wide measurement range: Detection of slight contacts of a few gram weight equivalent should be possible, but the sensor must also not be overdriven when moving or lifting weights in the range of 1–2 Kg.

Response dynamics: The sensor signal should have a rise time below 20 ms to allow the implementation of controlled grasping.

Reliability: A strong demand for the choice of the sensor is a proven sensor technology which affords little maintenance and has a sufficient life time.

Size and ease of integration: The sensor device should be small enough to fit into the phalanxes of the Karlsruhe Robot Hand [6], which is of the size of the human hand.

Electrical interface and measurement electronics: The sensor should provide an electrical interface with low cable count and that is not sensitive towards moderate electrical interference. The measurement electronics must be small in size and should offer a standard PC communication interface like RS232 or USB.

Beside these basic demands a further strong requirement for the tactile sensors is the capability to determine the contact normal force vector (CNFV) which allows for dextrous manipulation and reactive grasping with several common control algorithms.
For the tactile sensor system several sensor types have been investigated for their suitability. Sensors for force measurement may be divided in scalar and matrix type sensors. Using matrix force sensors the CNFV may be approximated from the measurement data when assuming a contact area larger than the sensors' grid resolution. Manufacturers of commercial matrix type force sensors are [3][2][4]. These sensors are usually manufactured as flexible sheets with uniformly distributed adjacent sensor cells. The cells usually have the shape of squares with the length of an edge ranging down to a few millimeters, defining the resolution. There is no off-the-shelf solution available for matrix force sensors in the application area of robot hands. The sensors always need to be customized in terms of geometry and resolution, which results in considerable costs for this type of sensor. All manufacturers examined require the customer to use a special sensor signal

processing hardware unit for processing sensor data. The sensor technology used for force sensors relies either on a variant of the FSR (Force Sensing Resistor) principle [3][4][1] or on the capacitive effect [2].
FSRs are also very common in tactile input devices like keyboards and keypads for PCs or hand-held devices. Sensor design and properties are in a technological mature state which allows immediate deployment of this sensor for tactile force and contact sensing to the required degree.
Although the term FSR itself is copyrighted by [1] the other manufacturers exploit material properties in a similar way, so that they are summarized under this term. The FSR sensor itself consists of two polymer sheets, one with a printed conducting electrode pattern and one with a semiconducting layer [7]. The resistance of the sensor decreases with increasing pressure applied to it as the two sheets are pressed together. The force sensing resistor is despite its name neither a pure force nor a pure pressure sensor. Its change in resistance is related to the portion of sensor surface addressed, the elasticity of the actuator and the electrode design on the sheet. The sensor element can be used as a pressure sensor if the force is applied to a major portion of the sensor area. This can be achieved by an appropriate actuator that distributes the force equally to the full sensor area. As a tactile sensor for a robot hand it is sufficient to overlay a spherical layer of an elastomer as actuator element on top of the sensor surface. The response time of an FSR is in the range of 1–2 ms, which is sufficient for tactile exploration and CNFV control.

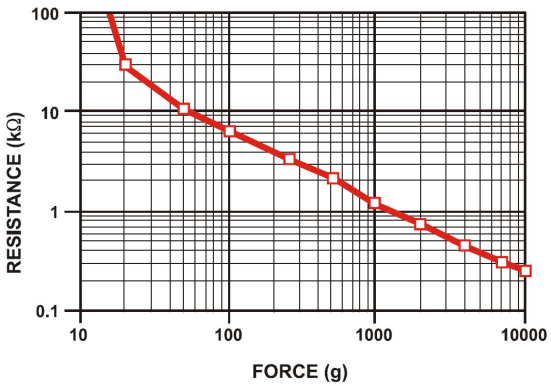


Figure 1: FSR Characteristics: Resistance vs. Force [1].

FSRs in general require a break force to switch from zero conductivity to a finite resistance value defining the beginning of the dynamic range. From here the characteristic usually follows an inverse power law as shown in the double logarithmic plot in figure 1.
This nonlinear resistance-force relation needs to be linearized by calibration if required for quantitative measurements.

2.2 Cursor Navigation Sensors for Tactile Sensing

Beside 1-D FSR elements for orthogonal contact force measurement the FSR technology is also used in cursor navigation devices as they are assembled in hand-held devices or laptop computers. These sensor devices have four FSR elements arranged in quadrants onto which the force is distributed by an actuator stick as shown in figure 2. The stick is covered with a rubber cap to provide friction during actuation.
When the actuator stick is bent the applied force is distributed and the resistance value of the four sensor elements change uniquely to the 2-D angle and the magnitude of the force.
After linearization of the resistance value for each quadrant the CNFV may be calculated as the resulting force from the four measurement values.

Figure 2: *MicroJoystick* input device [1].

A disadvantage of the sensor assembly is the actuator stick itself as it can not be integrated satisfactorily into a robot hand. It sticks out of the hand plane which makes it vulnerable against force overload. Also, the measurement range of the sensor is not sufficient towards smaller forces. Experiments have shown that the break force of the sensor device is the equivalent of approximately 70 gram weight. This is not sufficient to detect contact immediately or to control the CNFV with detailed resolution. Further more the electrical connection is only possible with a special flex board style connector that has to be glued with conductive adhesive to the contacts of the sensor, which increases the number of steps necessary for assembly.
Despite these issues the sensor may well be deployed to explore haptical features of objects in contact with the robot hand. This sensor was integrated into a two jaw gripper to investigate its properties and suitability for tactile exploration and reactive grasping, details will be shown later in this report.
Meanwhile also the mere four quadrant sensor element without actuator is available for purchase.
This FSR device is directly solderable to a PCB, alternatively copper wires may be soldered to the sensor. For proper operation the sensor needs to be used in conjunction with an actuator element. With this sensor it is sufficient to cover it with a spherical elastomer layer that distributes the applied force across the sensing area. The sensitivity can be adjusted by shape, thickness and elasticity of the elastomer layer. Experiments have shown that this sensor device can measure down to the force equivalent of 5 gram weight by overlaying a spherical silicone cap as shown in figure 3.
The upper limit of the measurement range is approximately the equivalent of 1.5 Kg.
The package of the *MicroNav 360* [1] sensors also make them a suitable base element for tactile matrix sensors as they can be arranged as grids on a carrier PCB.

2.3 Sensor Electronics and System Integration

By using a simple voltage divider circuit a measurable voltage signal can be generated from the changing resistance value of the FSR. The resistance value of the FSR ranges from 5 kΩ at high forces to approximately 100 kΩ at forces just above the break force point. The voltage signals of the sensors are directly connected to ADC inputs of a microcontroller. The microcontroller communicates to a host PC via a standard interface,

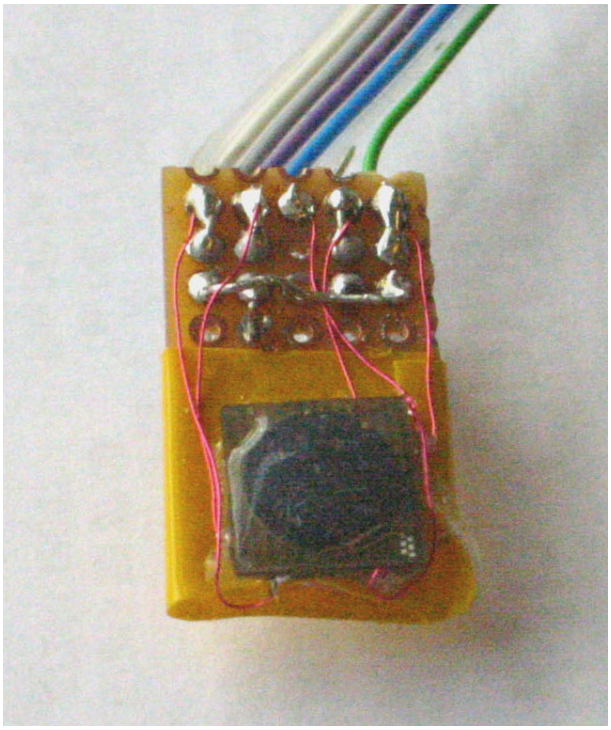


Figure 3: *MicroNav 360* with silicon cap and carrier board.

e.g. RS232, CAN or Bluetooth. This data acquisition circuit is generic and may be used for all types of FSR sensors. The microcontroller is used as a measurement unit and does not calculate a calibration function on the acquired measurement data. This is performed by software running on the connected PC for ease of software maintainability.

Figure 4 shows a jaw gripper equipped with cursor navigation sensors. Here four devices were soldered to a PCB to investigate the feasibility of a matrix sensor field.

The data acquisition circuit is attached to the backside of one jaw, also the typical flex board cables as needed for the *MicroJoystick* device [1] are visible. This setup is used as a demonstrator for investigating the characteristics of the *MicroJoystick* cursor navigation sensor in tactile exploration. As mentioned before the bulky geometry of the sensors actuator cap and the connection wires make this setup sensitive towards mechanical damage. Also, the grippers contact area is reduced to the actuator caps front surface which is not suitable for clamping objects.

In a second approach cursor navigation sensors with silicone actuator caps were integrated into a humanoid robot hand [6].

Figure 5 shows the FSR sensor element integrated into the thumb tip of the humanoid robot hand. The active sensor area is covered with a thin layer of silicone that was adapted to the shape of the underlying silicone finger tip. This silicone cap naturally flows to a spherical shape which results in a proper actuator for this sensor. An advantage of this design is that the finger surface area is not affected by the integration of the sensor and the stable mechanical design of the finger tips can be maintained.

First experiments approve this integrated tactile sensor a high sensitivity in the desired range from approximately 10 gram up to more than a Kilogram weight. Upcoming investigations will determine to what resolution this sensor design permits reliable tactile exploration and grasp control.

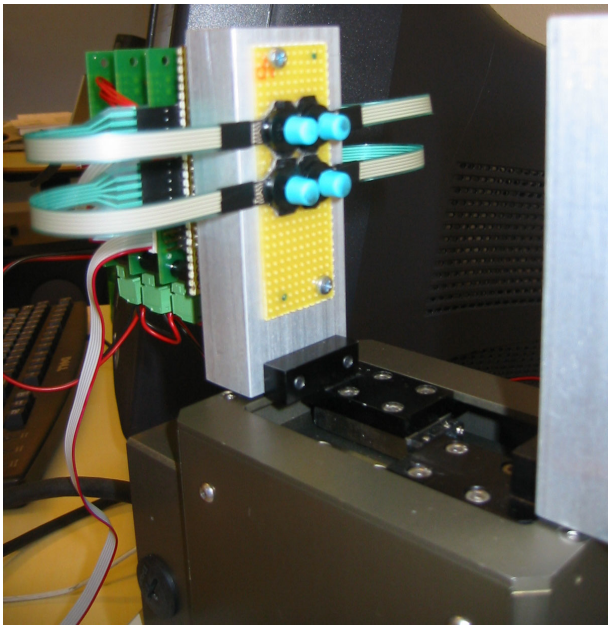


Figure 4: Four *MicroJoystick* devices mounted to a jaw gripper.

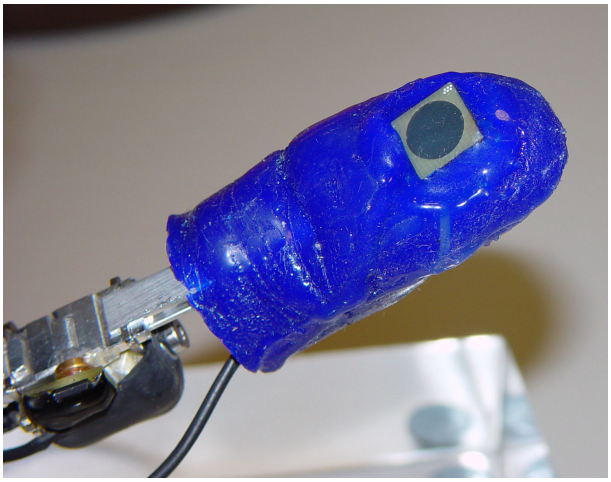


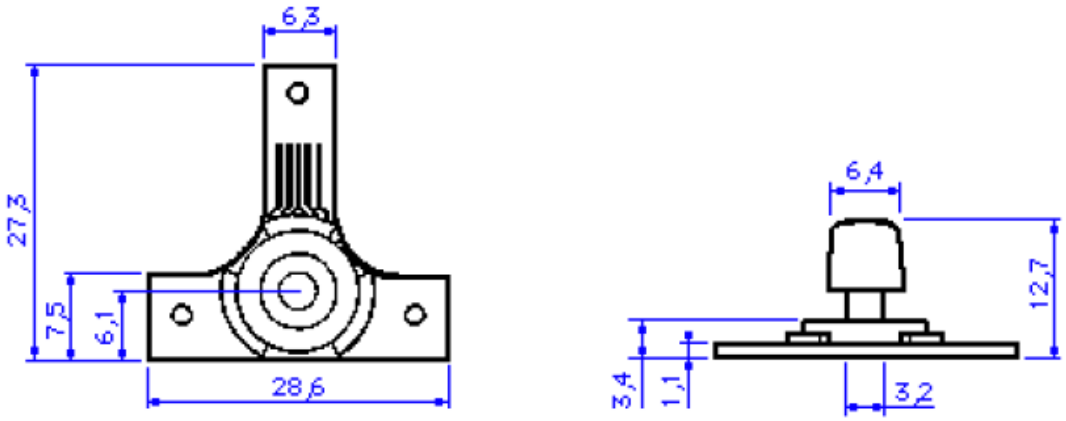
Figure 5: Integration of *MicroNav 360* sensor device .

3 Basic sensor properties

This chapter will go into more details of the properties and characteristics of the *MicroJoystick* sensor.

3.1 Physical properties

The sensor is mounted on a piece of PCB which acts as a base plate for further mounting, and also gives a pinout of the electric terminals from the four internal FSR sub-sensors. The dimensions of this base plate is shown in figure 6(a) and figure 6(b), and the pinout of the electric terminals can be seen in figure 2 as 5 traces.

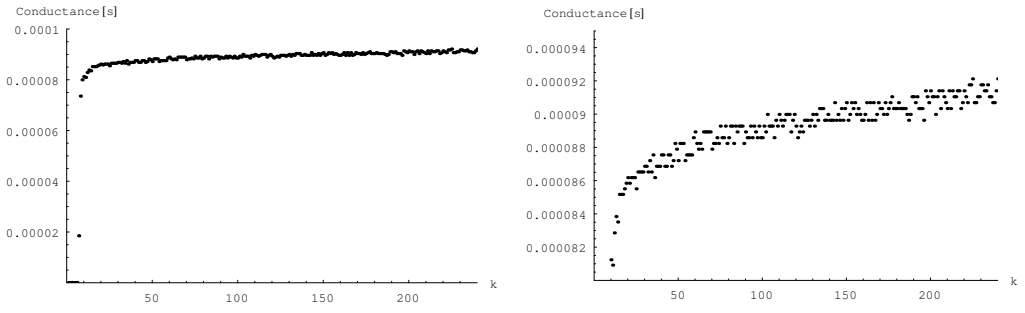


(a) (b)

Figure 6: MicroJoystick Sensor Dimensions [mm]. **(a)** Top View. **(b)** Side view

3.2 Step-response

Our first approach in analyzing the sensor was to get a step response by applying a normal force on the sensor simply by slowly bringing the sensor in contact with a flat surface and keeping it steady.



(a) (b)

Figure 7: Step response. k=sampling. Sampling time is 10Hz. **(a)** Full Range. **(b)** Magnification of the small force range.

The signal from each of the four sub-sensors was sampled with a frequency of 10Hz, and the measurements from one of the sub-sensors is plotted in figure 7(a). The moment the sensor makes contact with the surface

the signal grows rapidly. Afterwards the signal actually keeps growing slowly even if the sensor is kept in a constant position. Figure 7(b) gives a more detailed impression of the same measurements. The plot show the signal over a period of about 20 seconds,. Further tests have shown that the signal actually keeps growing for up to about 60 seconds.

3.3 Linearization

The purpose of linearization is to discover a function that can convert the measured sensor signal to a force value, within a suitable error interval. The input to this function will be the measured signal from the sensor. The physical property of the sensor that can be measured is the resistance of each FSR, which is converted into a voltage s in the interval 0–5 V using a voltage divider.
One possibility would be to directly translate this voltage into a force value using an approximated function f_v of the form in equation 1.

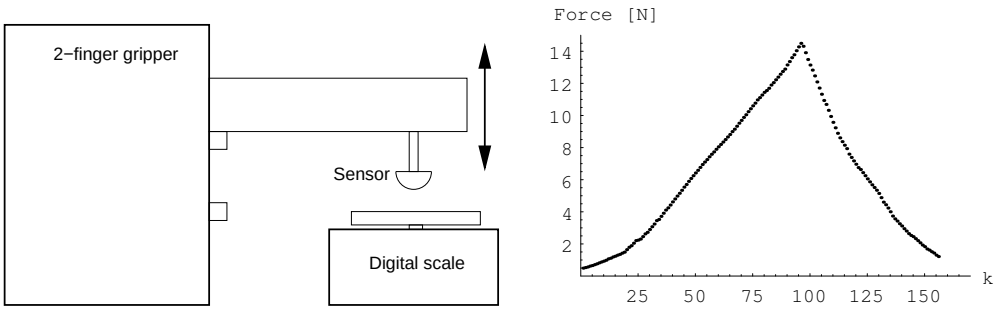
$$force = f_v(s) \tag{1}$$

This has the drawback that the resistence R of each sub-sensor is translated into a voltage in a non-linear way using the voltage-divider relationship. This means that we lose the linear relationship between the measured resistance and the force.
To take advantage of this expected linear property we could calculate the resistance from the voltage, and try to find a function that translates this into a force. Even better is to use the conductance, since it has a nice and close to linear relationship to the force. This gives a relation as shown in equation 2.

$$force = f_c(\frac{1}{R}) \tag{2}$$

Since the sensor is expected to have a linear relationship in the low force region, the function is expected to have the form shown in equation 3. For higher forces a higher order function might be needed to get a suitable approximation.

$$f_c(x) = \alpha + \beta \cdot x \tag{3}$$



(a) (b)

Figure 8: Linearization Experiment. **(a)** Setup for the linearization experiment. **(b)** Magnitude of the applied force during each sampling k.

To collect a number of measurements from the sensor with a known force applied, a digital scale was used to measure the force. The experiment was carried out by mounting a sensor on one of the fingers of a parallel gripper, so that the sensor was pointing downwards and could be moved up and down. The digital scale was

placed under the sensor, and by changing the position of the finger the sensor would apply a force on the scale which could be measured. This setup is shown in figure 8(a).
The sensor was moved closer to the scale in small steps. In each position the total force was measured with the digital scale, and the four reading from the four sub-sensors were recorded. This gave an increasing force on the sensor in each step. After a force of about 14 N was reached, the inverse experiment was made by slowly moving the sensor away and thereby decreasing the force in slow steps. This resulted in 156 measurements. The applied force measured with the scale is shown in figure 8(b).
The force was applied as a normal force, so we would expect equal readings from all four sub-sensors. The mathematical distribution of the force is shown in figure 9. Since the sensor consists of four sub-sensors, the force should be equally distributed over these so that each will measure a force of $F_i = \frac{1}{4}F$, where $F_i \in \{F_N, F_S, F_E, F_W\}$.

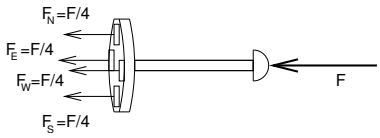


Figure 9: Distribution of force on the sub-sensors when a normal force is applied

3.3.1 Experiment Data

Figure 10 shows the results from the North sub-sensors. The force on the x-axis is $\frac{1}{4}$ of the total force measured with the scale, since this should be the theoretical magnitude of force that is applied to one sub-sensor. The red marks are the results from the first part of the experiment when the force was increasing, and the blue marks are from the last part where the force was decreasing. During the last part of the experiment the sensor seems to have higher readings than when the force is increasing. It was shown that the sensor readings would keep increasing when a force was applied to the sensor over a period of time, and it seems to be the same characteristic that is the reason for the different readings in figure 10. During the experiment the sensor was subject to a high force, and this large force would make the sensor readings higher during the last part of the experiment.

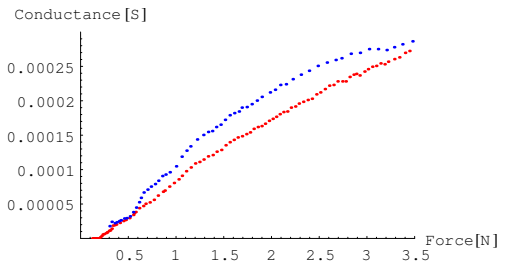
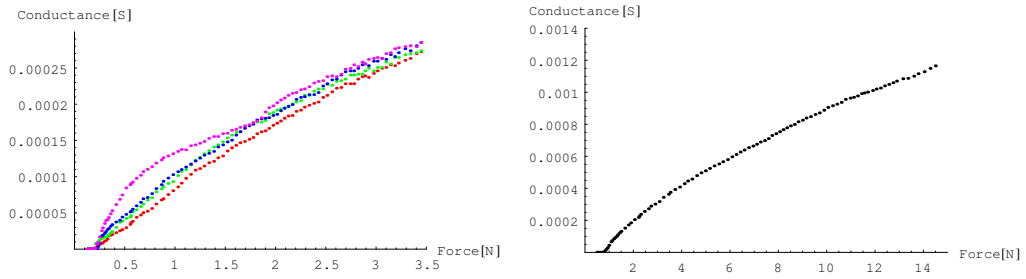


Figure 10: Measured conductance from the north sub-sensor during linearization experiment. **Blue:** Decreasing force. **Red:** Increasing force

Figure 10 shows that the relationship doesn't follow a straight line over the measured range, but more a second order relationship.
In the low-force range it would be possible to approximate the relationship with a first order function, which might not give the same accuracy as a second order but would be preferred for the following reasons:

- Individual sensor calibration is easier and requires less data points.
- FSR sensors have low accuracy, so the loss of precision by using a 1st order relationship does not make a significant difference.

Since the characteristic of FSR sensors differs very much from part to part, an easy way to calibrate individual sensors would probably give a better accuracy than an uncalibrated sensor with a higher order approximation. And since forces as high as the experiment tested (up to 14 N) will not be needed in the current application, it would be better to linearize the sensors in the low force range using a first order relationship.



(a) (b)

Figure 11: Measured conductance from all sensors during linearization experiment with only increasing force. **(a)** Conductance from each sub-sensors. Red=North, Blue=South, Green=East & Purple=West. **(b)** Sum of the conductance from all sub-sensors.

The data points from the part of the experiment with increasing force from all four sub-sensors are shown in figure 11(a). The graph representing the W sub-sensor, is growing clearly faster than the other 3 sub-sensors, which is an unexpected behavior. The other 3 sub-sensors have a slight difference in measured force, with the red graph (North), lying a little lower than the last two: blue (South) and green (East). Since the force was applied as a normal force, equal readings from all 4 sub-sensors would have been expected. The reason for the differences could be:

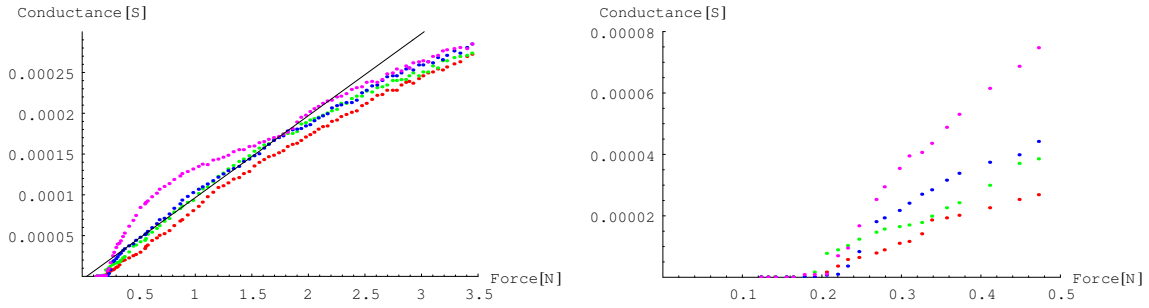
- Differences in the characteristics of each individual FSR sub-sensor
- The force was not exactly normal, but had a tangential component due to surface friction

The difference in the characteristics for each sub-sensor could be overcome by calibrating each sub-sensor individually, by assuming equal distribution of force on each sub-sensor, and then approximating the force-conductance relationship individually for each sensor.
But several other experiments of the same kind showed that it was not the same sub-sensor that gave the highest reading each time. This shows that the differences was due to a tangential force acting on the sensor-tip, and tilting the actuator slightly. Since an tangential force will tilt the actuator, it will apply a torque on the sensor. This torque will give a force acting on the N and S sub-sensor with the same magnitude, but in different directions. The same holds for the W and E sensor. This means the effect of the tangential force could be minimized by using the sum of the 4 sub-sensor readings. This sum is plotted in figure 11(b) against the total force.
Because of the intended application datapoints above 8 N will be disregarded in the following. The remaining datapoints are approximated to a straight line using least-squares approximation. This gives a

force-conductance relationship as shown in equation 4. The inverse which should be used to calculate the force from a measured conductance is shown in equation 5.

$$C(f) = -3.9336 \cdot 10^{-6} + 100.5 \cdot 10^{-6} \cdot f \quad (4)$$

$$F(c) = 0.0391384 + 9949.88 \cdot c \quad (5)$$



(a) (b)

Figure 12: Measured conductance from all sensors during linearization experiment with only increasing force. Red=North, Blue=South, Green=East & Purple=West. **(a)** Result of linearization compared to the measured results. **(b)** Magnification of the low force range shows the magnitude of the break force.

In figure 11(a) you also see the result of the break force required to get readings from the sensor. With very low forces applied there is no reading from the sensor. This situation is more clearly shown in figure 12(b) which is a magnification of the small force range of figure 11(a). This shows that the break force for each sub-sensor is approximately $0.2N$, which means that a minimum force of $0.8N \approx 80g$ is required if it is applied as a normal force, and the force is equally distributed over the 4 sub-sensors.

3.4 Sensor Model

To better understand and interpret the signals from each sensor, we have created a theoretical and simplified model of the sensor. It can be used to calculate the expected signal from the sensor when a known force is applied and vice versa.
To simplify the model we have made the following assumptions:

1. Each sub-sensor measures a force in only one point
2. Each sub-sensor measures only the perpendicular component of the force in this point
3. The contact point is always assumed to be the point at the end of the tip
4. A tangential force will contribute to a torque around a fixed rotation point

To calculate how much a force vector applied at the end of the tip contributes to the internal force sensors, we split the force-vector into the normal and tangential component. The normal component is shown in figure 13(a). Since the contact point is assumed to be in the middle of the sensor-tip, the normal component will be equally distributed over the four sub-sensors.

The tangential component is a little more complicated. Since the sub-sensors only measure a normal force, they do not directly detect the tangential component. But according to assumption three, this force will contribute to a torque around a fixed point, which is assumed to be in the middle of the sensor. The distance from the tip to this center of rotation is defined as *l1*, and depends on the geometry of the sensor. Since the sub-sensors only measures the force in one single point, this torque will give a force in each sub-sensor depending on the distance from the center of rotation to the location if the sub-sensor. This distance is defined as *l2*. This is shown in figure 13(b).

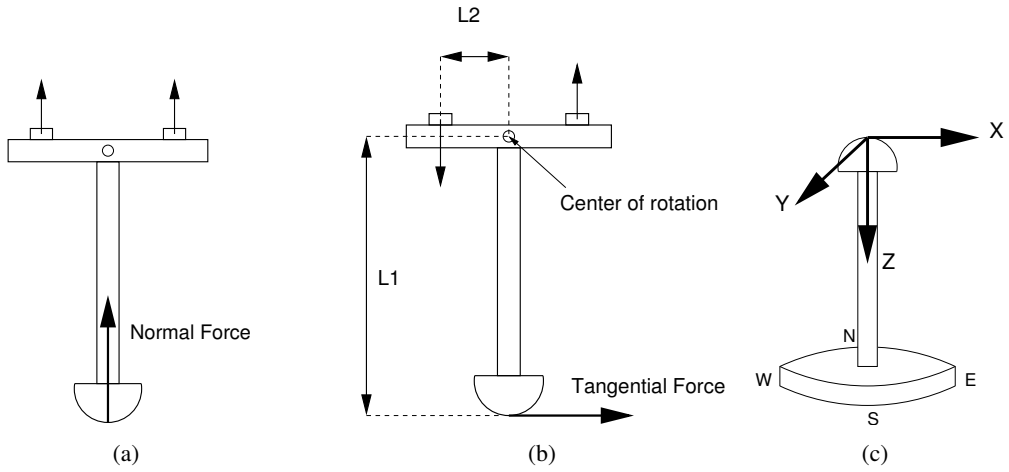


Figure 13: Distribution between the subsensors of a force applied to the sensor tip. **(a)** Force normal to the sensor **(b)** Force tangential to the sensor **(c)** Definition of the sensor frame

3.4.1 From force to sensor measurements

We assume the applied force vector is specified in the frame of the sensor which is defined with the origin at the tip of the sensor, the z-axis pointing towards the base and the x-axis in the direction of the East sub-sensor as shown in figure 13(c).
If the force vector is defined as $(x, y, z)^T$, the normal component is simply the z-component. The contribution to each of the 4 sub-sensors from the normal component is shown in equation 6.

$$n_{normal} = s_{normal} = e_{normal} = w_{normal} = \frac{1}{4}z \tag{6}$$

The Y-component results in a torque that is measured by the N and S sub-sensor, and the X-component gives a torque measured by the E and W sub-sensors. For example is the torque due to the X-component of the force is $T = l1 \cdot x$, and the magnitude of the force measured by the E sub-sensor due to this torque $E_{tangent} = \frac{1}{l2} \cdot T = \frac{l1}{l2} \cdot x$. I define the relation ship $\frac{l1}{l2}$ as α which gives the results shown in vector form in equation 7.

$$\begin{pmatrix} n \\ s \\ e \\ w \end{pmatrix}_{tangent} = \alpha \cdot \begin{pmatrix} -y \\ y \\ x \\ -x \end{pmatrix} \tag{7}$$

The total force measured by each sub-sensor is shown in equation 8.

$$\begin{pmatrix} n \\ s \\ e \\ w \end{pmatrix} = \alpha \cdot \begin{pmatrix} -y \\ y \\ x \\ -x \end{pmatrix} + \frac{1}{4}z \tag{8}$$

3.4.2 From sensor measurements to force

The method to calculate the expected sensor output due to a known force is informal. Since the force vector consists of 3 unknowns and the sensor measurements gives 4 known values the inverse calculation should also be possible.
We define the 4 dimensional vector consisting of the measured sub-sensor forces as $\mathbf{U}$, the applied 3D force vector as $\mathbf{f}$ and the theoretical sub-sensor forces shown in equation 8, which depends on the applied force as $\mathbf{V}(\mathbf{f})$. The goal is to find the vector $\mathbf{f}$ which minimizes the error between $\mathbf{U}$ and $\mathbf{V}$ as defined in equation 9.

$$E = ||\mathbf{U} - \mathbf{V}(\mathbf{f})|| \tag{9}$$

Differentiating E with respect to x, cancels out all terms of y and z as shown in equation 10. The same happens when differentiating with respect to y and z. Solving these 3 equations for zero gives the force vector that minimizes the error. This is shown in equation 11.

$$\delta\frac{E}{\delta x} = 2\alpha(-e + w + 2\alpha x) \tag{10}$$

$$\mathbf{f} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \frac{e-w}{2\alpha} \\ \frac{s-n}{2\alpha} \\ e+n+s+w \end{pmatrix} \tag{11}$$

3.5 Sub-conclusion

It was observed that the readings from the sensor keep growing if a constant force is applied. The same behavior was seen when a decreasing force was applied after the sensor had been subject to a very high force. This large settling time seems to be a characteristic of these types of sensors. If a force is applied in a short period, for example for detecting a surface normal, this large settling time might have little effect. But if the force is kept for a long period, for example when grasping and holding an object, it might be required to compensate for this effect.
The sensor output was linearized for forces below $8N$ using the measurements obtained before the high force was applied, and thus with minimum error from the settling effect. The result was a first order function.
A very simple model of the sensor able to calculate the expected sensor readings with a known force was derived. The same model was used to derive the opposite calculation, where the applied force could be calculated from the sensor readings. This model is very simple and based on many assumptions. It is unclear how precise it will be in practice and it has not been verified yet. But it could act as a base for further development of a more complex sensor model, which also take into consideration the location of the contact point.

4 Object Property Detection Experiments

One of the planned applications of the sensors is to explore the shape of an object, or in the simple case to detect the normal direction of a surface. A set of experiments was carried out to investigate whether these

two sensors were able to measure the normal of a surface by a single touch, and how precise the direction can be measured. Additionally two experiments were carried out to explore the ability of the *MicroJoystick* sensor to detect the weight and elasticity of an object beeing grasped.

4.1 Definition of Surface direction

The direction of the surface in the experiments is defined relative to the orientation of the sensor itself. The direction of a surface is normally defined as a 3D vector, normal to the surface. This is not very useful in this case because the sensor might not be equally good at detecting how much the surface is tilted and in what direction it is tilted.

For this reason we define the orientation of the surface with two angles, the tilt angle α and the roll angle β.

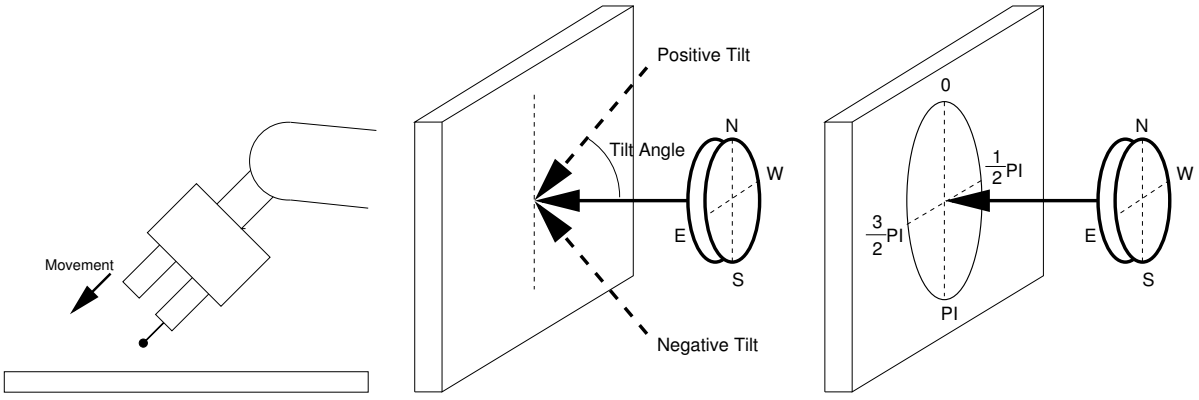
Figure 14(b) shows the definition of the tilt angle. A tilt angle of $\alpha = 0$ means the sensor is normal to the surface, and a positive tilt angle means the sensor is tilted towards the N direction.

Figure 14(c) shows the direction of the roll angle. A roll angle of $\beta = 0$ gives a positive tilt in the N direction, $\beta = \frac{\pi}{2}$ gives a tilt in the W direction and so on.

Using the ranges of the α and β angles in equation 12 and 13 covers all possible orientations of a surface relative to the sensor when the sensor is pointing in the direction of the surface.

$$-\frac{\pi}{2} \leq \quad \alpha \quad \leq \frac{\pi}{2} \tag{12}$$

$$0 \leq \quad \beta \quad < \pi \tag{13}$$



(a) (b) (c)

Figure 14: **(a)** Direction of the robot movement in the experiments. **(b)** The tilt angle used to define a surface orientation. **(c)** The roll angle used to define a surface orientation.

4.2 Definition of Sensor Measurements

The measurement from each sensor is a 4 dimensional force vector $\vec{S}$ consisting of the force measurement from each sub-sensor, shown in equation 14.

$$\vec{S} = \begin{pmatrix} n \\ s \\ e \\ w \end{pmatrix} \tag{14}$$

$$\hat{S} = \frac{\vec{S}}{|\vec{S}|} \tag{15}$$

$$\vec{D} = \begin{pmatrix} n-s \\ e-w \end{pmatrix} \tag{16}$$

$$\vec{P} = \begin{pmatrix} n-s \\ e-w \end{pmatrix} \cdot \frac{1}{|\vec{S}|} \tag{17}$$

Since the purpose of the sensor is to measure the orientation of a surface, we are interested in the difference between the opposite forces in each axis. We call the 2 dimensional vector representing this difference $\vec{D}$. It is defined in equation 16.
Since the contact between the sensor and the surface was done by placing the sensor in a steady position, the force applied by the robot arm to the sensor is unknown. The force can vary depending on how fast the robot was able to stop the movement after a contact was detected, the magnitude of the threshold force used to detect a contact and the velocity of the movement towards the surface.
To overcome this problem we assume that the relationship between the individual sub-sensor force measurements is constant for a given surface orientation, even if the total applied force is varied. By scaling the vector $\vec{S}$ into a unit vector, it is thus possible to compare the sensor reading independent of the magnitude of the applied force. The vector consisting of the difference in each axis of the unit vector $\vec{P}$ is defined as shown in equation 17. The vector $\vec{P}$ will be used to detect the surface normal.

4.3 Surface Normal Detection Experiments

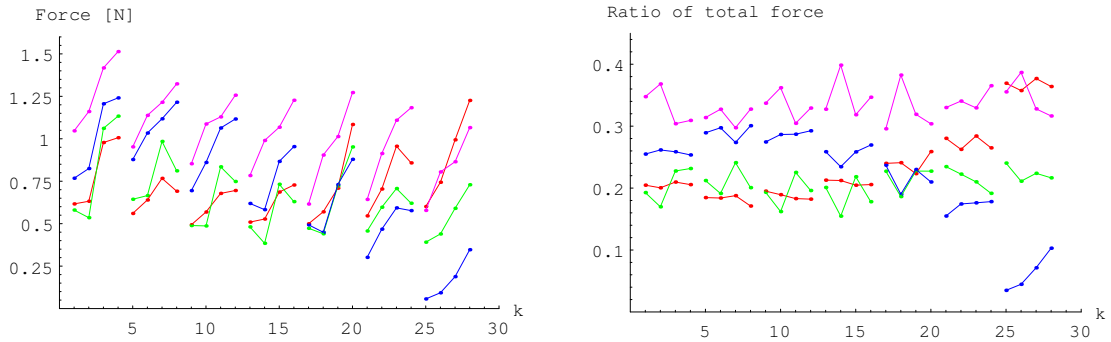
4.3.1 Experimental setup

For this experiments one sensor was mounted on a robot finger. The surface used in the experiment was a wooden plate with plastic lamination. This gave a smooth surface, and some friction with the rubber tip of the sensor.
The orientation and position of the surface was known, and the sensor was moved in position above the surface, and repeatedly moved in contact with the surface from different directions. The movement towards the surface was made in a slow movement, following a straight line in 3D space in the direction parallel to the sensor, and keeping constant orientation as seen in figure 14(a). The movement had a velocity of approximately 1–2 mm per second. As soon as a force above a given threshold was detected from the sensor, the movement was stopped, and the position held for one second. Then the sensor was moved away from the surface again.
The following experiments were carried out:

1. Difference in applied force using the *MicroJoystick* sensor
2. Variations of tilt angle using the *MicroJoystick* sensor
3. Variations of roll and tilt angle using the *MicroJoystick* sensor
4. Variations of roll and tile angle using the *MicroNav* sensor

4.3.2 Experiment 1: Difference in applied force using the *MicroJoystick* sensor

In section 4.2 it was assumed that by scaling the measurement vector $\vec{S}$ to unit length, the result was independent of the total force applied. The first experiment was made to verify the correctness of this, and to investigate how the measurements changed when the total applied force was varied.
This was done by creating contact with a surface from the same direction four times. The first time the movement was stopped when a force of 0.2 N was measured, the next time when 0.4 N was measured, 0.6 N and 0.8 N. This gave four different reading with the same surface orientation but different magnitude of applied force. The experiment was repeated seven times with different tilt angles in the range from $0 \leq \alpha < \frac{\pi}{4}$ and a roll angle of $\beta = 0$.
The values of $\vec{S}$ from each measurement are plotted as colored dots in figure 15(a). The measurements connected with a line were made with the same tilt angle. It is clear to see that the force values in general are higher in the fourt measurement in each group, than in the first.
The values of the unit vector $\hat{S}$ are plotted the same way in figure 15(b) where the y-axis now corresponds to the force relative to the total force. The expected result would be that all values in the same group would have the same value, since they were done with the same tilt angle.
The values are not exactly the same, but it is still an improvement compared to figure 15(a). The standard deviation of all 28 groups of measurements were calculated to 0.018.
This is the expected standard deviation within the measurements when the magnitude of the total force is varied. Since the total sum of all four sensor values in the $\hat{S}$ vector is 1.00, a standard deviation of 0.018 equals 1.8 % which is a small error compared to other factors.



(a) (b)

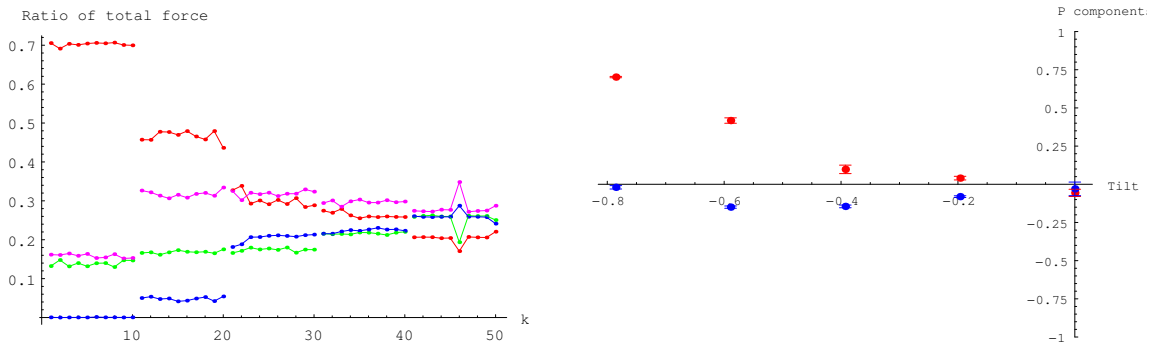
Figure 15: Results from experiment 1. k is the number of the measurement. Measurements done with the same tilt angle are connected with lines. Red=North, Blue=South, Green=East & Purple=West. **(a)** Measurements shown as force values. **(b)** Measurements shown as the ratio of the total force.

4.3.3 Experiment 2: Variations of tilt angle using the *MicroJoystick* sensor

The next experiment was made to investigate whether the sensor is able to detect variations in the tilt angle. The roll angle was kept constant at $\beta = 0$, and the sensor was moved into contact with a surface with the five tilt angles shown in eq. 18. For each tilt angle the experiment was repeated 10 times. This resulted in 50 measurements.

$$\alpha \in \left\{0,\ -\frac{1}{16}\pi,\ -\frac{2}{16}\pi,\ -\frac{3}{16}\pi,\ -\frac{4}{16}\pi\right\} \tag{18}$$

Since β is zero and α negative, the sensor was tilted in the direction of the S sub-sensor. See figure 14(b).



(a) (b)

Figure 16: Results from experiment 2. **(a)** Measurements shown as the ratio of the total force. k is the number of the measurement. Measurements done with the same tilt angle are connected with lines. Red=North, Blue=South, Green=East & Purple=West. **(b)** Mean values of $\vec{P}$ shown for the tested tilt angles. Red=North/South axis, Blue=East/West axis.

The result is plotted in figure 16(a). The mean of the 10 contacts done with each angle is listed in table 1. The standard deviation from the mean in the 10 contacts are listed in table 2. The average value of the standard deviations is 0.00886.
This standard deviation seems to be very small, which can also be seen by the almost horisontal lines in figure 16(a).

Tilt	**N**	**E**	**S**	**W**
$-\frac{4}{16}\pi$	0.702	0.139	0	0.158
$-\frac{3}{16}\pi$	0.466	0.168	0.048	0.318
$-\frac{2}{16}\pi$	0.302	0.174	0.205	0.319
$-\frac{1}{16}\pi$	0.264	0.216	0.223	0.297
0	0.204	0.253	0.26	0.283

Table 1: Results from the experiment with variations of the tilt angle. Each value is the mean value of the 10 measurements done.

Tilt	**N**	**E**	**S**	**W**
$-\frac{4}{16}\pi$	0.0046	0.007	0.0004	0.0049
$-\frac{3}{16}\pi$	0.0139	0.0039	0.0046	0.0084
$-\frac{2}{16}\pi$	0.0176	0.0048	0.0108	0.0076
$-\frac{1}{16}\pi$	0.008	0.0026	0.0048	0.0052
0	0.0126	0.0212	0.0111	0.0233

Table 2: The standard deviation in the measurements from the experiment with variations of the tilt angle.

To try to detect the tilt angle, we looked at the difference in each axis defined as the vector $\vec{P}$ in section 4.2. This difference is listed in table 3(a), which was calculated by using the mean of the measurements from table 1. The standard deviation of the measurements from the mean is listed in table 3(b). The unit is of the same type as table 1 so the values can be directly compared. These values are plotted in figure 16(b). The sensor was tilted in the S-direction, which can be seen by a growing magnitude of the $N - S$ value. The $E - W$ would be expected to be constant zero.

Table 3: Result from experiment with variations in tilt angle. **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	N-S	E-W
$-\frac{4}{16}\pi$	0.702	-0.02
$-\frac{3}{16}\pi$	0.417	-0.15
$-\frac{2}{16}\pi$	0.098	-0.145
$-\frac{1}{16}\pi$	0.04	-0.081
0	-0.056	-0.03

(b)

Tilt	N-S	E-W
$\frac{4}{16}\pi$	0.004	0.011
$\frac{3}{16}\pi$	0.018	0.008
$\frac{2}{16}\pi$	0.028	0.009
$\frac{1}{16}\pi$	0.012	0.006
0	0.024	0.044

4.3.4 Experiment 3: Variations of both roll and tilt angle using the *MicroJoystick* sensor

The experiment in section 4.3.3 was done with a constant roll angle of $\beta = 0$. This experiment will try to explore the behavior of the sensor when the roll angle is varied.
This was done by applying the tilt angles listed in eq. 19, with three different roll angles listed in eq. 20. Five measurements were taken for each angle, to be able to calculate the mean and the variance.

$$\alpha \in \left\{0,\ -\frac{1}{32}\pi,\ -\frac{2}{32}\pi,\ -\frac{3}{32}\pi,\ -\frac{4}{32}\pi, -\frac{5}{32}\pi,\ -\frac{6}{32}\pi,\ -\frac{7}{32}\pi,\ -\frac{8}{32}\pi\right\} \tag{19}$$

$$\beta \in \left\{0,\ \frac{1}{6}\pi,\ \frac{2}{6}\pi\right\} \tag{20}$$

The tilt angles are divided into smaller intervals than in the last experiment. It was not possible to apply tilt smaller than $-\frac{5}{32}\pi$ when a roll angle of $\frac{1}{6}\pi$ or $\frac{2}{6}\pi$ was used. The reason for this is that the corner of the robot finger would hit the surface, so a higher tilt angle would require a different geometric design of the sensor setup.
The two components of the $\vec{P}$ vector with a roll angle of $\beta = 0$ are listed in table 4(a). Each of these values is the mean of five measurements done with the same angle. The standard deviations of these measurements are listed in table 4(b). These values are plotted in figure 17(a).

Table 4: Part of the result from experiment 3 with a roll angle of $\beta = 0$ **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	N-S	E-W
$\frac{8}{32}\pi$	0.68	-0.038
$\frac{7}{32}\pi$	0.547	-0.042
$\frac{6}{32}\pi$	0.343	-0.149
$\frac{5}{32}\pi$	0.113	-0.15
$\frac{4}{32}\pi$	0.07	-0.153
$\frac{3}{32}\pi$	0.04	-0.093
$\frac{2}{32}\pi$	0.029	-0.077
$\frac{1}{32}\pi$	0.051	-0.04
$\frac{0}{32}\pi$	0.008	-0.049

(b)

Tilt	N-S	E-W
$\frac{8}{32}\pi$	0.006	0.013
$\frac{7}{32}\pi$	0.011	0.013
$\frac{6}{32}\pi$	0.009	0.008
$\frac{5}{32}\pi$	0.017	0.002
$\frac{4}{32}\pi$	0.026	0.009
$\frac{3}{32}\pi$	0.014	0.006
$\frac{2}{32}\pi$	0.029	0.009
$\frac{1}{32}\pi$	0.015	0.041
$\frac{0}{32}\pi$	0.013	0.02

Looking at the plot, the result looks similar to the results found in the previous experiment. The only difference between the two experiments was smaller step size in the tilt angle. With these smaller steps it is

more clearly to see that the value of $N - S$ start growing significantly when the sensor is tilted more than $0.4\,rad$ (about 22°). For smaller tilt angles the behavior seems to be more random.
The same measurements, but with a roll angle of $\beta = \frac{1}{6}\pi$ are listed in table 5(a), and the standard deviations are listed in table 5(b). The values are plotted in figure 17(b). The expected result would be that the $E - W$ value is growing in the negative direction as the tilt angle becomes higher (seen from right to left in the plot) since the sensor is tilted slightly in the E-direction. Since the sensor is also tilted in the S-direction the value of $N - S$ should also be growing although less than in the last experiment. Looking at figure 17(b) it is clear to see that this behavior is not quite as expected. The values are close to zero, changing sign randomly and the variance of the measurements is large compared to the values themselves. It is important to notice that the x-axis only goes down to -0.5 where as in figure 17(a) it goes down to -0.8, because the hand was unable to tilt further in this setup.

Table 5: Part of the result from experiment 3 with a roll angle of $\beta = \frac{1}{6}\pi$ **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$-\frac{0}{32}\pi$	0	-0.067
$-\frac{1}{32}\pi$	-0.029	-0.073
$-\frac{2}{32}\pi$	0.005	-0.003
$-\frac{3}{32}\pi$	0.014	0.048
$-\frac{4}{32}\pi$	0.07	0.024
$-\frac{5}{32}\pi$	0.08	-0.153

(b)

Tilt	**N-S**	**E-W**
$-\frac{0}{32}\pi$	0.003	0.014
$-\frac{1}{32}\pi$	0.008	0.019
$-\frac{2}{32}\pi$	0.013	0.007
$-\frac{3}{32}\pi$	0.007	0.038
$-\frac{4}{32}\pi$	0.013	0.056
$-\frac{5}{32}\pi$	0.02	0.05

The means from the experiment with a roll angle of $\beta = \frac{2}{6}\pi$ are listed in table 6(a), and the standard deviations listed in table 6(b). The result is plotted in figure 17(c). The expected result is a slightly growing value of $N - S$ and a growing value of $E - W$ in the negative direction, as the sensor is tilted. The result does not show this behavior. Instead the values seem to be close to zero with a high variance.

Table 6: Part of the result from exeriment 3 with a roll angle of $\beta = \frac{2}{6}\pi$ **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$-\frac{0}{32}\pi$	-0.007	-0.05
$-\frac{1}{32}\pi$	-0.02	-0.038
$-\frac{2}{32}\pi$	-0.037	0.028
$-\frac{3}{32}\pi$	-0.025	-0.014
$-\frac{4}{32}\pi$	-0.032	-0.049
$-\frac{5}{32}\pi$	0.031	0.002

(b)

Tilt	**N-S**	**E-W**
$-\frac{0}{32}\pi$	0.009	0.013
$-\frac{1}{32}\pi$	0.002	0.023
$-\frac{2}{32}\pi$	0.009	0.037
$-\frac{3}{32}\pi$	0.018	0.074
$-\frac{4}{32}\pi$	0.029	0.074
$-\frac{5}{32}\pi$	0.052	0.058

4.3.5 Experiment 4: Variations of both roll and tilt angle using the *MicroNav* sensor

The surface normal experiment were carried out again using the *MicroNav* sensor. For these experiments we used the same rubber tip that were originally mounted on the MicroJoystick sensor. This was glued to the *MicroNav* sensor as seen in figure 18. This makes it possible to get into contact with a surface even if it is not completely normal to the finger. To explore whether this new sensor design is able to measure the

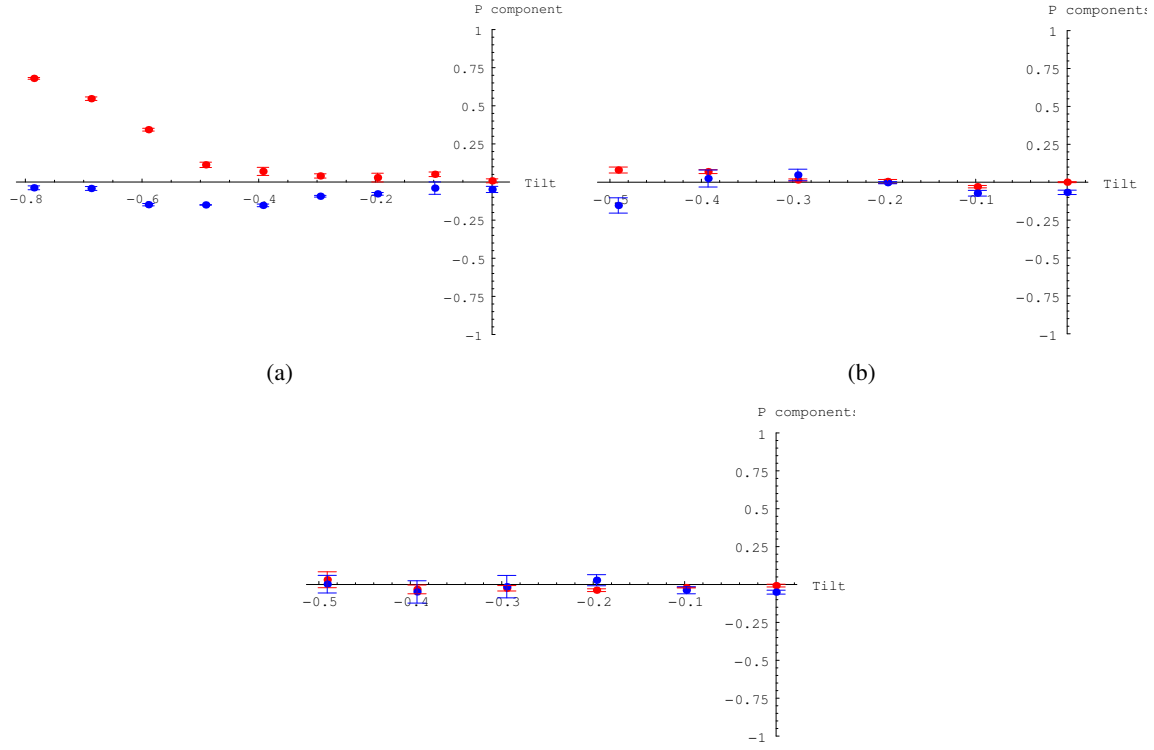


Figure 17: Results from experiment 3. Mean values of $\vec{P}$ shown for the tested tilt angles. Red=North/South axis, Blue=East/West axis. **(a)** For a roll angle $\beta = 0$. **(b)** For a roll angle $\beta = \frac{1}{6}\pi$. **(c)** For a roll angle $\beta = \frac{2}{6}\pi$.

normal direction of a surface, we tested it with a series of contact angles. All possible pairs of the following roll and tilt angles were tested:

$$\alpha \in \left\{0,\ \frac{1}{32}\pi,\ \frac{2}{32}\pi,\ \frac{3}{32}\pi,\ \frac{4}{32}\pi,\ \frac{5}{32}\pi,\ \frac{6}{32}\pi\right\} \tag{21}$$

$$\beta \in \left\{0,\ \frac{1}{6}\pi,\ \frac{2}{6}\pi,\ \frac{3}{6}\pi\right\} \tag{22}$$

Each angle pair was tested six times to make it possible to measure both the mean value and the standard deviation. The difference in the two axes for a roll angle of $\beta = 0$ is shown in table 7(a) and the standard deviation of the measurements are shown in table 7(b). A roll angle of $\beta = 0$ means the sensor is tilted in the direction of the north sub-sensor. The results are also shown in figure 19(a), where the values have been substracted with the values from a tilt angle of $\alpha = 0$. In practise this is done by calibrating the sensor by touching a surface directly normal to the sensor, and using these readings as zero.
The rest of the measurements with different roll angles, and the standard deviation of the measurements are listed in table 8(a)-10(b). These values are shown graphically in figure 19(b), 19(c) and 19(d). A roll angle of $\beta = \frac{1}{2}\pi$ means the sensor is tilted in the direction of the west sub-sensor.
These results shows that the measured values from the sensor depends on both the roll and the tilt angle. A higher tilt angle gives a higher difference in the force readings for that respective axis which depends on the roll angle. It seems to be growing linear up to a point about $0.4\,rad$ ($22°$) where the values seems to be

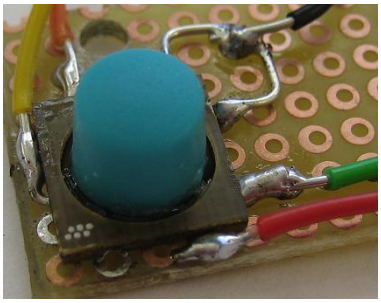


Figure 18: The MicroNav sensor with a mounted sensor tip.

become lower even when the tilt angle is higher. This shows that it should be possible to measure the roll and til angle for tilt angles lower than about 22° using the *MicroNav* sensor.

Table 7: Results from surface experiment with the *MicroNav* sensor using a roll angle of $\beta = 0$. **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.106	-0.206
$\frac{1}{32}\pi$	0.2	-0.211
$\frac{2}{32}\pi$	0.35	-0.196
$\frac{3}{32}\pi$	0.439	-0.229
$\frac{4}{32}\pi$	0.495	-0.219
$\frac{5}{32}\pi$	0.478	-0.235
$\frac{6}{32}\pi$	0.417	-0.254

(b)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.009	0.0358
$\frac{1}{32}\pi$	0.042	0.0306
$\frac{2}{32}\pi$	0.0291	0.0303
$\frac{3}{32}\pi$	0.0436	0.0266
$\frac{4}{32}\pi$	0.03	0.0104
$\frac{5}{32}\pi$	0.0127	0.0063
$\frac{6}{32}\pi$	0.0125	0.0094

4.3.6 Discussion of Results

Experiment 2 shows that there is a good repeatability in the information received from the sensor when applied to a surface. Each orientation of the surface was touched 10 times with the sensor, and the average standard deviation of the measurements was only about 0.9% of the total force applied to the sensor. Experiment 1 shows that the repeatability is acceptable, even when the magnitude of the total force was varied. The average standard deviation in this case was found to be about 1.8% of the total force applied. In this experiment the variance in the measurements was provoked to be greater by changing the force on purpose, so this should be a worst case scenario.

Experiment 2 and 3 shows it should be possible to detect the tilt angle using the *MicroJoystick* sensor when the sensor is tilted with more than about 22°. For lower angles the values from the sensor are small compared to the deviation, so the measurements are very noisy. This noise could possibly be due to the friction force, which acts on the sensor tip in the tangential direction of the normal. Since the sensor was unable to reach the high tilt angles in experiment 3, it is unfortunately not possible to conclude how good the sensor is at detecting the roll angle.

Experiment 4 shows the *MicroNav* sensor can be used to detect the roll and tilt angle of a surface if the tilt angle is lower than about 22°.

Table 8: Results from surface experiment with the *MicroNav* sensor using a roll angle of $\beta = \frac{1}{6}\pi$. **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.106	-0.206
$\frac{1}{32}\pi$	0.203	-0.216
$\frac{2}{32}\pi$	0.32	-0.335
$\frac{3}{32}\pi$	0.418	-0.338
$\frac{4}{32}\pi$	0.422	-0.388
$\frac{5}{32}\pi$	0.44	-0.405
$\frac{6}{32}\pi$	0.354	-0.388

(b)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.009	0.0358
$\frac{1}{32}\pi$	0.016	0.0407
$\frac{2}{32}\pi$	0.0465	0.0319
$\frac{3}{32}\pi$	0.0242	0.0263
$\frac{4}{32}\pi$	0.0404	0.0169
$\frac{5}{32}\pi$	0.0338	0.047
$\frac{6}{32}\pi$	0.0128	0.0279

Table 9: Results from surface experiment with the *MicroNav* sensor using a roll angle of $\beta = \frac{2}{6}\pi$. **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.106	-0.206
$\frac{1}{32}\pi$	0.15	-0.321
$\frac{2}{32}\pi$	0.246	-0.436
$\frac{3}{32}\pi$	0.282	-0.483
$\frac{4}{32}\pi$	0.294	-0.458
$\frac{5}{32}\pi$	0.304	-0.492
$\frac{6}{32}\pi$	0.307	-0.507

(b)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.009	0.0358
$\frac{1}{32}\pi$	0.0533	0.0055
$\frac{2}{32}\pi$	0.0218	0.0237
$\frac{3}{32}\pi$	0.0351	0.046
$\frac{4}{32}\pi$	0.0223	0.0417
$\frac{5}{32}\pi$	0.0687	0.0523
$\frac{6}{32}\pi$	0.0257	0.074

Table 10: Result from surface experiment with the *MicroNav* sensor with a roll angle of $\beta = \frac{3}{6}\pi$. **(a)** Mean values of $\vec{P}$ **(b)** Standard deviation in the measurements of $\vec{P}$

(a)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.106	-0.206
$\frac{1}{32}\pi$	0.082	-0.29
$\frac{2}{32}\pi$	0.211	-0.407
$\frac{3}{32}\pi$	0.211	-0.518
$\frac{4}{32}\pi$	0.22	-0.577
$\frac{5}{32}\pi$	0.238	-0.593
$\frac{6}{32}\pi$	0.222	-0.582

(b)

Tilt	**N-S**	**E-W**
$\frac{0}{32}\pi$	0.009	0.0358
$\frac{1}{32}\pi$	0.0159	0.0664
$\frac{2}{32}\pi$	0.0211	0.0203
$\frac{3}{32}\pi$	0.0364	0.058
$\frac{4}{32}\pi$	0.0324	0.0246
$\frac{5}{32}\pi$	0.0485	0.0558
$\frac{6}{32}\pi$	0.0295	0.0603

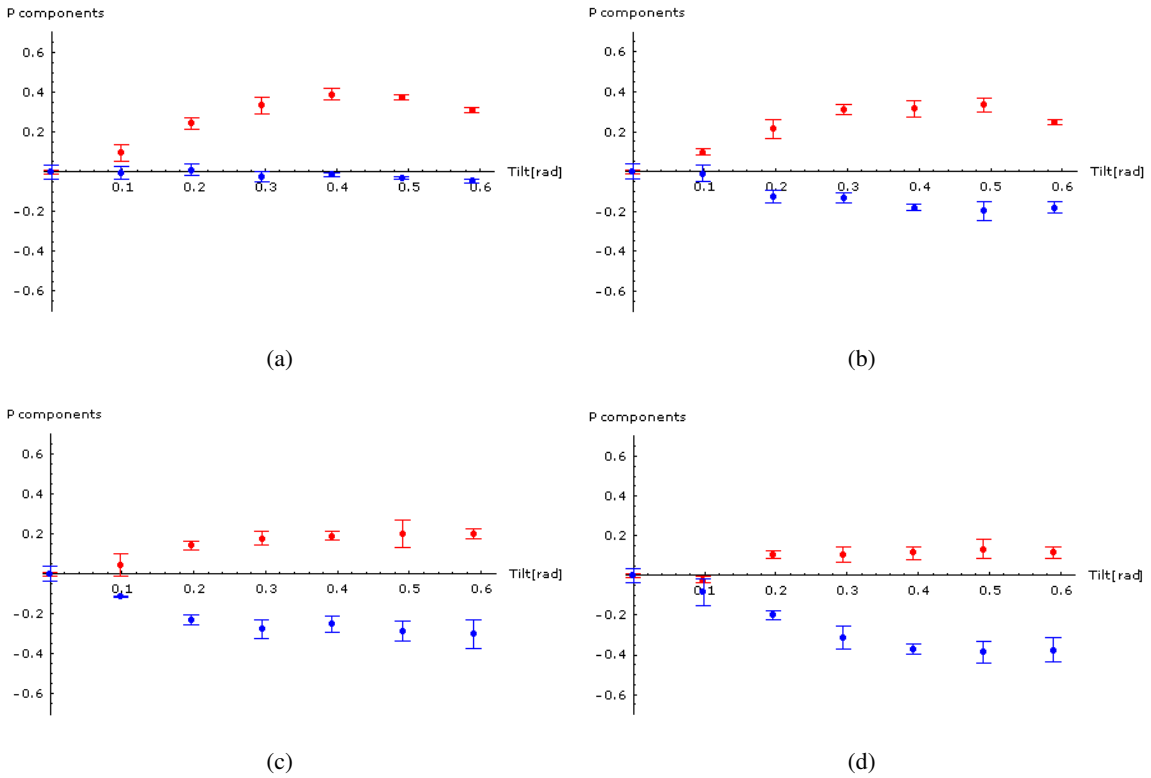
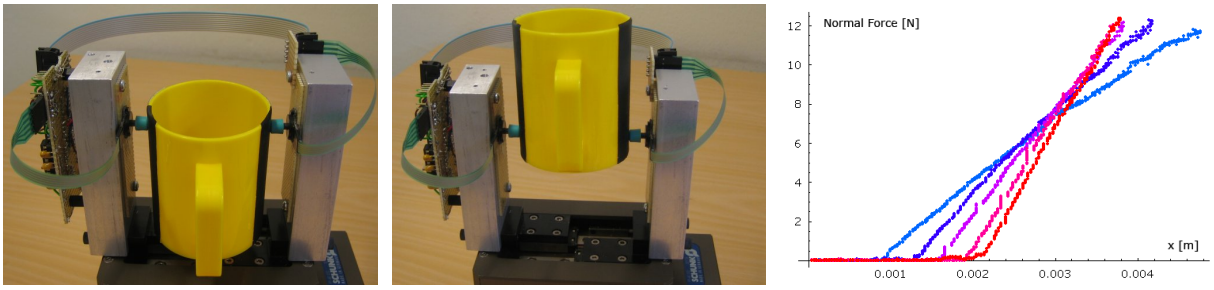


Figure 19: Results from surface normal experiment using the *MicroNav* sensor. Mean values of $\vec{P}$ shown for the tested tilt angles. Red=North/South axis, Blue=East/West axis. **(a)** For a roll angle $\beta = 0$. **(b)** For a roll angle $\beta = \frac{1}{6}\pi$. **(c)** For a roll angle $\beta = \frac{2}{6}\pi$. **(d)** For a roll angle $\beta = \frac{3}{6}\pi$.

4.4 Elasticity Experiment

The elasticity of an object or a surface is the ability to deform when a force is applied to a contact point, for example during a grasp. The elasticity is a useful property to be known in grasping, since it makes it possible to predict in what way the object will deform during a grasp. It is also a relavant object property. A high elasticity makes an object for example useless to be used in a hammer like way. The ability of the sensor to measure the elasticity of an object was explored using a two sensor setup mounted on a parallel gripper. The gripper would close around a plastic cup with the two sensors as the only contact points. When the cup was grasped in the top it would deform into an oval shape, since it is more flexible at this point (see figure 20(a)). In the lower part of the cup the shape was stabilized by the bottom of the cup, and would not easily deform (see figure 20(b)). The parallel gripper was closed slowly with a constant velocity and stopped when a certain maximum force was reached. The diameter of the cup at the top was a little larger than at the bottom. It is 59 mm at the top versus 57.5 mm at the bottom. The experiment was repeated five times with different contact locations. The force measured by one of the sensors as the gripper was closing can bee seen in figure 20(c). The light blue graph shows the result when the contact point was at the top of the cup, and the red graph shows for the bottom of the cup. The rest of the graphs show the contact points in between these two.

It can clearly be seen that the force is growing slowly when grasping at a soft location, and growing fast when grasping at a hard location. The sensor is also able to detect the different diameter of the cup, de-



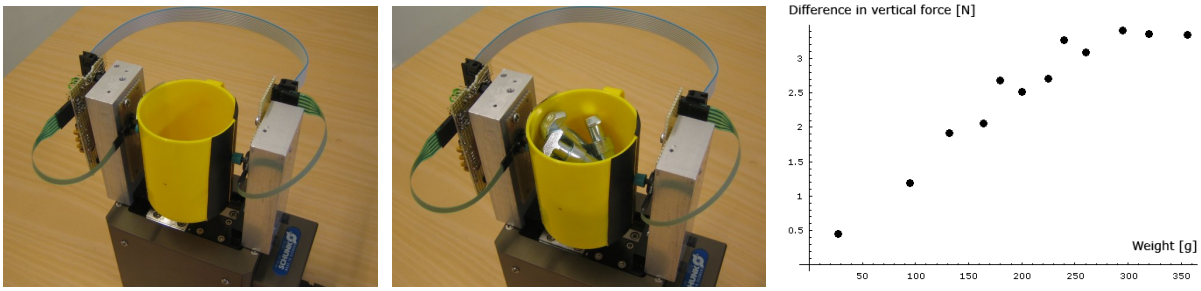
(a) (b) (c)

Figure 20: Elasticity Experiment Setup **(a)** The cup grasped at the top. **(b)** The cup grasped at the bottom. **(c)** Results from the experiment. Light Blue = grasp located at the top, Red = grasp located at the bottom, Other colors = grasp located in between

pending on the grasping point. When grasping at the top the sensor measures a contact after about 1 mm of movement, and when grasping at the bottom it measures a contact after about 2 mm of movement because of the smaller diameter.

4.5 Weight Experiment

The sensors are not expected to be able to measure the precise weight of an object, but we find it useful to investigate whether they are able to give an indication of the weight of a grasped object to attach properties such as 'fullness' or 'emptyness'. For this experiment the same sensor setup was used, and the sensor readings were recorded while objects with constant shape but different weights were grasped. The gravitational force would exert a downwards force on the object, so the weight was expected to me measurable in the force readings in vertical axis of the sensor. The plastic cup was grasped one time where it was empty (see figure 21(a)) and several times filled with different amounts of metal objects (see figure 21(b)).



(a) (b) (c)

Figure 21: Weight Experiment Setup **(a)** Empty Cup. **(b)** Full Cup. **(c)** Results from the experiment.

The result can be seen in figure 21(c). The y-axis shows the average of the force difference measured in the vertical direction of the two sensors. This value seems to depends linear on the weight of the object except for weights higher than about 300 g, where it seems to approach a limit.

5 Multisensorial surface exploration

This experiment shows that the sensor can be used to verify whether a surface predicted by the vision system actually exists. The visual prediction of surfaces is described in [5]. The scene chosen for verification consists of the closed white box placed on a black surface (see figure 22(a)). The vision system predicts three possible surfaces in the scene. A surface on the top of the box, a surface on the side of the box and a surface between the edge of the box and the edge of the black surface (see figure 22(b)).
Each of these three detected surfaces consist of a group of visual mono primitives each describing among other things the position of a point on the surface and the surface normal in that point. Using this information from a visual mono it is possible to create a movement where the sensor is moved through a point on the prodicted surface in a linear movement parallel to the surface normal. The verification of a surface is done by choosing one of more mono primitives on the surface for verification.
Figure 22(c) shows a situation where a visual mono on the top of the box has been chosen. The robot moves the gripper in position above the surface, and then does a straight line movement through the surface. When a contact is detected by the sensor (see figure 22(d), the movement is stopped and a haptic mono primitive is added at the point of contact. This haptic contact can now be used to measure the normal direction of the surface and other surface properties in that single point.
The top surface of the box was verified three times, resulting in three haptic mono primitives which can be seen in figure 22(e). Each haptic primitive is shown as a red square marking the position, and a green line marking the surface normal.
The vision system also predicted a surface located between the edge of the box and the edge of the black surface. To verify this surface a visual mono was chosen and the robot did the same straight line movement though the predicted surface. Since no surface exist in the point the robot moves through the predicted surface without detecting a contact and stops when a maximum distance has been reached (see figure 22(f)). Since the visual mono primitives on the same surface are grouped together it is now possible to remove all the monos from this wrongly predicted surface.

6 Conclusion

We have shown that it is possible to extract object information such as surface normal, weight and elasticity using the *MicroJoystick* and *MicroNav* sensors from Interlink electronics. By this we have demonstrated the usability of these sensors for tasks such as haptic exploration and grasping.

Acknowledgment

This work is supported by the PACO-PLUS project.

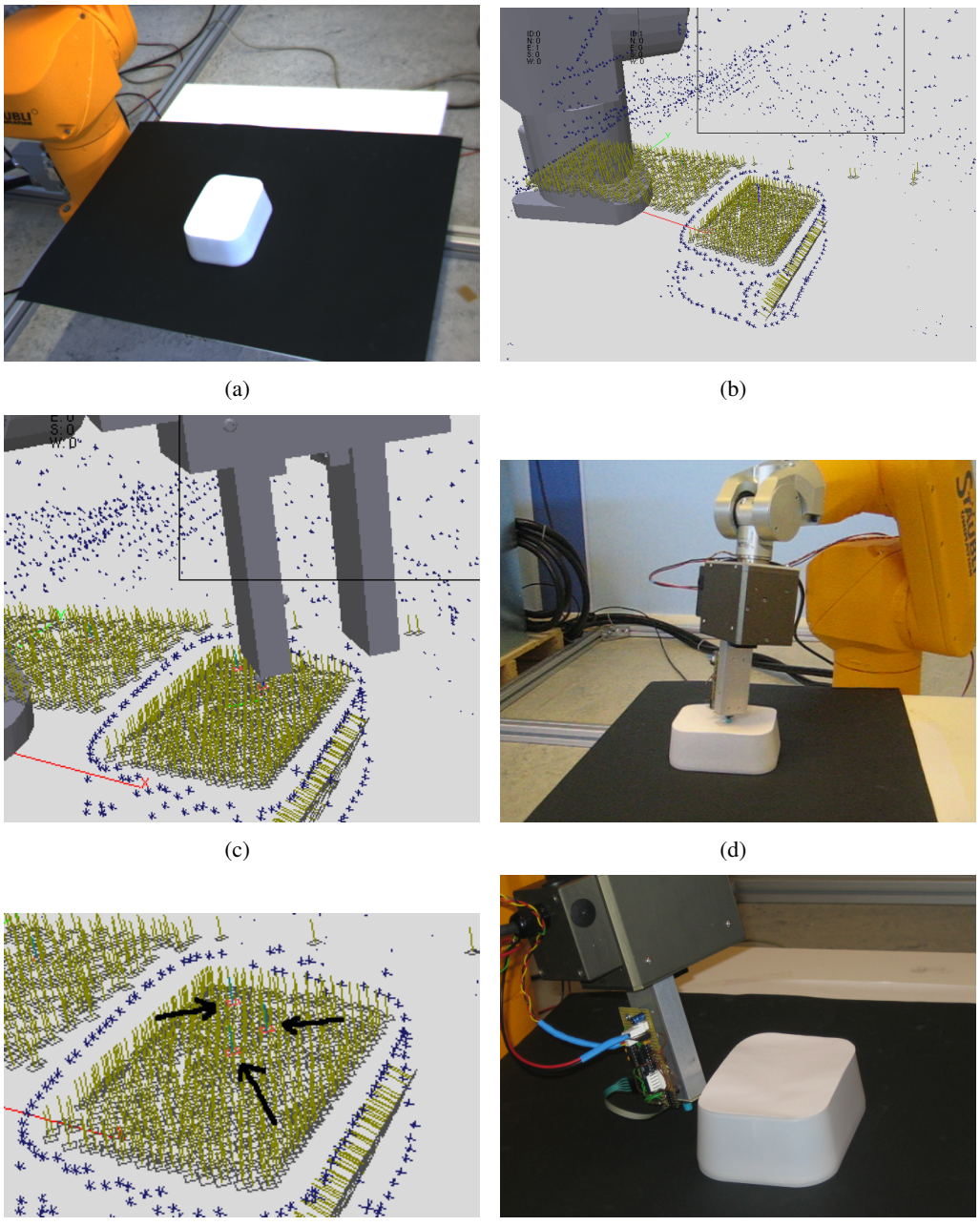
References

[1] Interlink electronics. http://www.interlinkelec.com.

[2] Pressure Profile Systems Inc. http://www.pressureprofile.com.

[3] Tekscan. http://www.tekscan.com.

[4] Weiss robotics. http://www.weiss-robotics.de/.



(a) (b) (c) (d) (e) (f)

Figure 22: Surface Verification Experiment. **(a)** Setup of the scene. **(b)** View of the 3 predicted surfaces. **(c)** The robot moving in position to verify the surface on the box. **(d)** The sensor in contact with the surface on the box. **(e)** The 3 detected haptic primitives shown as small red squares. **(f)** The robot moving through the wrongly predicted surface without detecting a contact.

[5] S. Kalkan, F. Wörgötter, and N. Krüger. Depth prediction at homogeneous image structures. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-2, 2007.

[6] S. Schulz, C. Pylatiuk, and G. Bretthauer. A new ultralight anthropomorphic hand. In *Internat.Conf.on Robotics and Automation 2001, Seoul, Korea*, 2001.

[7] Stuart I. Yaniger. Force Sensing Resistors: A Review of the Technology. *Electro International*, pages 666–668, 1991.

Robotics Group
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Technical Report no. 2007 – 4

Multi-modal Primitives: Local, Condensed, and semantically rich visual Descriptors and the Formalisation of contextual Information

Norbert Krüger, Nicolas Pugeault and Florentin Wörgötter

January 23, 2007

Title	Multi-modal Primitives: Local, Condensed, and semantically rich visual Descriptors and the Formalisation of contextual Information
Author(s)	Norbert Krüger, Nicolas Pugeault and Florentin Wörgötter
Publication History	

1 Introduction

There exists a large amount of evidence that the human visual system in its first cortical stages processes a number of aspects of visual data (see, e.g., [19, 39]). These aspects, in the following called visual modalities, cover, e.g., local orientation [19, 20], colour [20], junction structures [46], stereo [3] and optic flow [20]. At the first stage of visual processing (called 'early vision' in [29]), these modalities are computed locally for a certain retinal position. At a later stage (called 'early cognitive vision' in [29]), results of local processing become integrated with the spatial and temporal context. Computer vision has dealt to a large extent with these single modalities and in many computer vision systems, one or more of the above–mentioned aspects are processed in the first stages (see, e.g., [35, 45, 33]).
An important problem, the human visual system as well as any artificial visual system has to cope with, is the high degree of ambiguity and noise in these low level modalities that is unresolvable by local processes only. Reliable actions require a more stable representation of visual features. As a consequence, a disambiguation process that makes use of contextual information is needed. In [32] we have described two main regularities in visual data (that are also well recognised in the computer vision community) that underlie such an disambiguation process: (i) Coherent motion of rigid bodies and (ii) statistical interdependencies underlying most grouping processes. These two regularities allow to make predictions between locally extracted visual events and thereby to verify the spatio–temporal coherence of hypotheses.
The establishment of such a disambiguation process presupposes communication of temporal and spatial information. An efficient condensation of the locally extracted information implies:

Property 1. *The condensed information vector should allow for rich predictions between related (e.g. the change of position and appearance of a local patch under a rigid body motion) visual events;*

and

Property 2. *The condensed information vector need to reduce the dimensionality of the local signal to allow the process to work with limited bandwidth.*

In [25] it is argued that the need for properties 1 and 2 naturally result in symbolic representations. In this work, we present a novel kind of scene representation based on local symbolic descriptors that we call visual primitives (see figure 1).[1] In these primitives different visual modalities become combined in one local feature descriptor (section 2 and 3) that allows for the representation of visual scenes in a condensed way (satisfying property 2).
Furthermore, the primitives allow for rich predictions (property 2) since we can formulate efficiently statistical dependencies as operating in most perceptual grouping mechanisms as well as the change of image structure under a coherent motion (see section 5). Hence, locally computed primitives work as first guesses in a disambiguation process that is described in [41].
Our scene representation based on multi–modal primitives addresses a number of issues in an original way:

Multi–modality: Primitives cover the main visual modalities established in computer– and human vision and, hence, carry a rich semantic interpretation that facilitates the disambiguation process.

Condensation: Although primitives reduce the dimensionality of the image data, the significant aspects of image information are kept. For example, by using the primitives, we were able to achieve a stereo matching performance similar to correlation based methods that use the full image information (see [28]).

[1] A possible biological equivalent of the primitives are so called hyper–columns in the visual cortex (for a discussion, see [30]).

Dynamic Positioning and Completeness: The primitives semantically describe the image information in terms that are meaningful for image and scene understanding. This is achieved by dynamic search for primitives position resulting in localised symbolic descriptors that preserve a complete representation of structures. Namely, we have a description of contours, corners and surfaces and their mutual relations. We will show that in the case of contours this semantic extends naturally to 3D space (see 4.1).

Different Experts for different Structures: The interpretation of the local signal by the primitives is not static but depends on the intrinsic signal structure leading to a system of different experts for different signal structures such as edges, lines, homogeneous patches and corners (as also established in the human system).

Primitives Initialise Disambiguation: The primitives are not understood as a final statement about the local structure of a scene but a confidence associated to each primitive as well as its parameters as well become modified in disambiguation processes formalising contextual information. This paper is the first technical description of our visual primitives that have been applied already in various contexts (see, e.g., [31, 28, 22]). The primitives make use of a rather complex body of signal processing methods associated to the different visual modalities. Some of these aspects have been published earlier (such as e.g., the monogenic signal [14], a continuous concept of intrinsic dimension [27]) and are described briefly in this paper to make the presentation self-contained.
The system processes information over multiple stages (for an overview see figure 1) described in the following sections. In section 2, we will describe the processing of the individual modalities by linear and non–linear filtering processes. In section 3, we describe the condensation process generating primitives. In section 4, stereo–pairs of primitives are used to reconstruct information about the scene structure into *3D–primitives*. In section 5, we briefly describe the application of our primitives in an early cognitive architecture integrating perceptual grouping and motion as well as in the context of vision based robotics. A more detailed description the application of the primitive representation resulting in reliable and precise scene representations is given in [41].

2 Analysis of the local Signal Structure

In section 2.1 we will first describe how we distinguish different kinds of local image structures. The processing of the modalities orientation, phase and optic flow is then described in section 2.2 and 2.3. The results of the process described in this section are illustrated in a compact way in figure 1b).

2.1 Intrinsic Dimension

Different kinds of image structures coexist in natural images: homogeneous image patches, edges, corners, textures. Furthermore, certain concepts are only meaningful for specific classes of image structures. For example, the concept of orientation is well defined for edges or lines but not for junctions, homogeneous image patches or for most textures.
As another example, the concept of position is different for a junction as compared to an edge or an homogeneous image patch — see figure 2. a) in homogeneous areas of the image no particular location can be defined, and therefore an equidistant sampling is appropriate. b) For a line or edge structure the position can be defined using energy maxima. However, because of the aperture problem, this energy maxima will span a one–dimensional manifold, and therefore the feature can be localised only up to this manifold. This result in a fundamental ambiguity in the localisation of edge/line local features. c) At the contrary, the locus of a junction can be unambiguously defined by the point of line intersection (see figure 2c).
Similar considerations are required for other modalities such as colour, optic flow and stereo (see below).

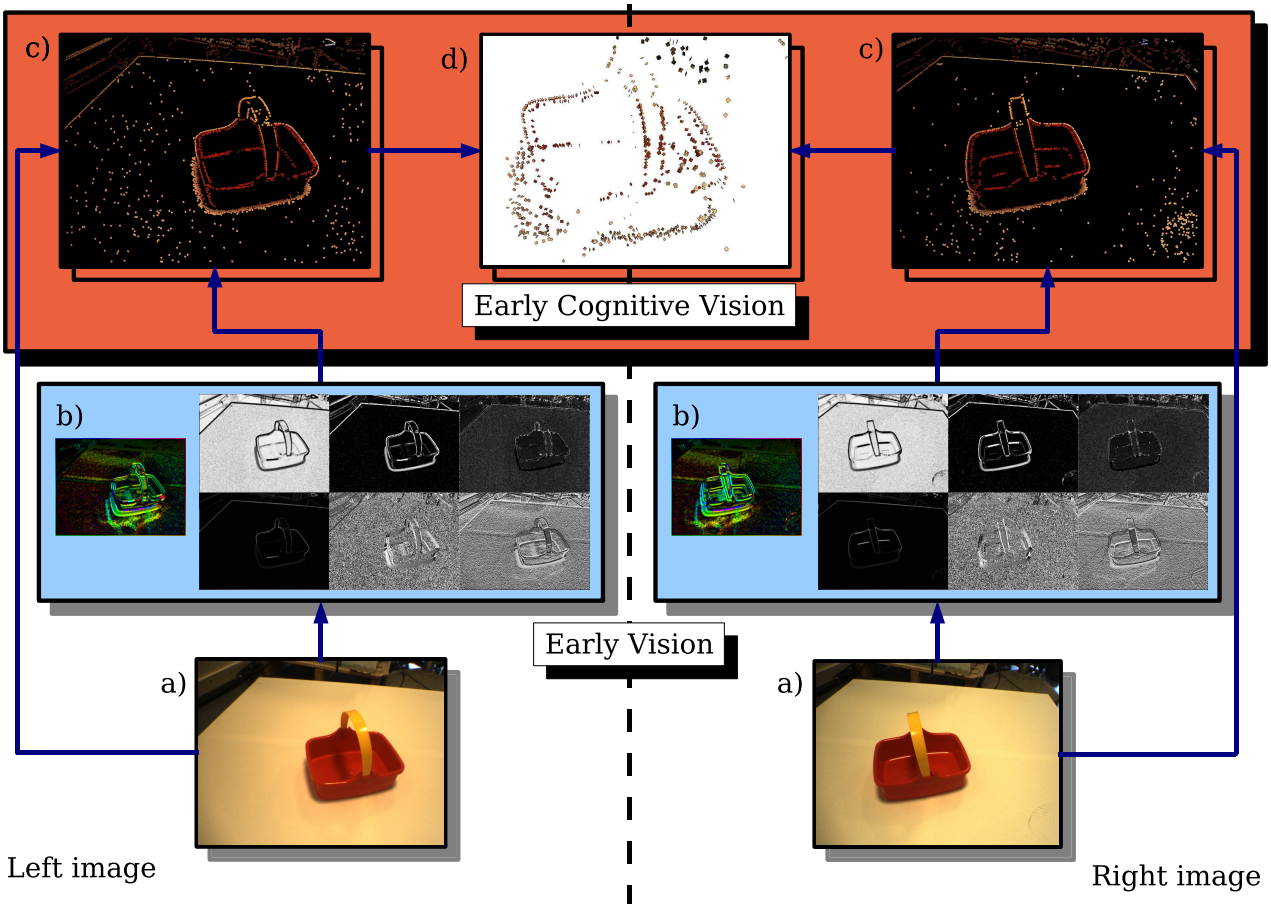


Figure 1: Overview of the primitive extraction scheme. **a)** a stereo–pair of images obtained from a pre–calibrated stereo rig. Therefrom, Early Vision processes are computed as shown in **b)**: the left image shows the optical flow extracted using the — see section 2.3. The hue of the pixels indicate that the orientation of the optic flow at this pixel is towards the margin of similar hue, and the intensity illustrate the magnitude of the flow vector); the bottom row of images shows the magnitude, orientation and phase of the signal — see section 2.2— from left to right respectively; The upper row shows the i0D, i1D and i2D confidences — see section 2.1 — from left to right respectively. In all those graphs the intensity encodes the strength of the filter response (white for high, black for low). In **c)** the information from the Early Vision module is combined in a sparse, condensed way into the Early Cognitive Vision module — see section 3. The image shows the primitives extracted from the images shown in a) **d)** these primitives are then matched across the two stereo-views and the correspondences thereof allows to reconstruct 3D–primitives, that extend naturally the primitive information to 3D space — see section 4.

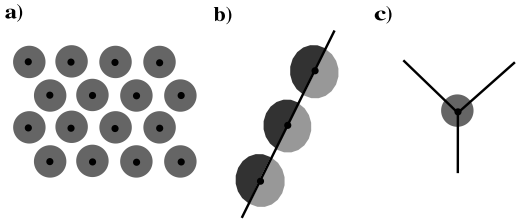


Figure 2: the different localisation problems faced by the different classes of image structure: a) homogeneous area; b) edge or line; and c) junction (see text).

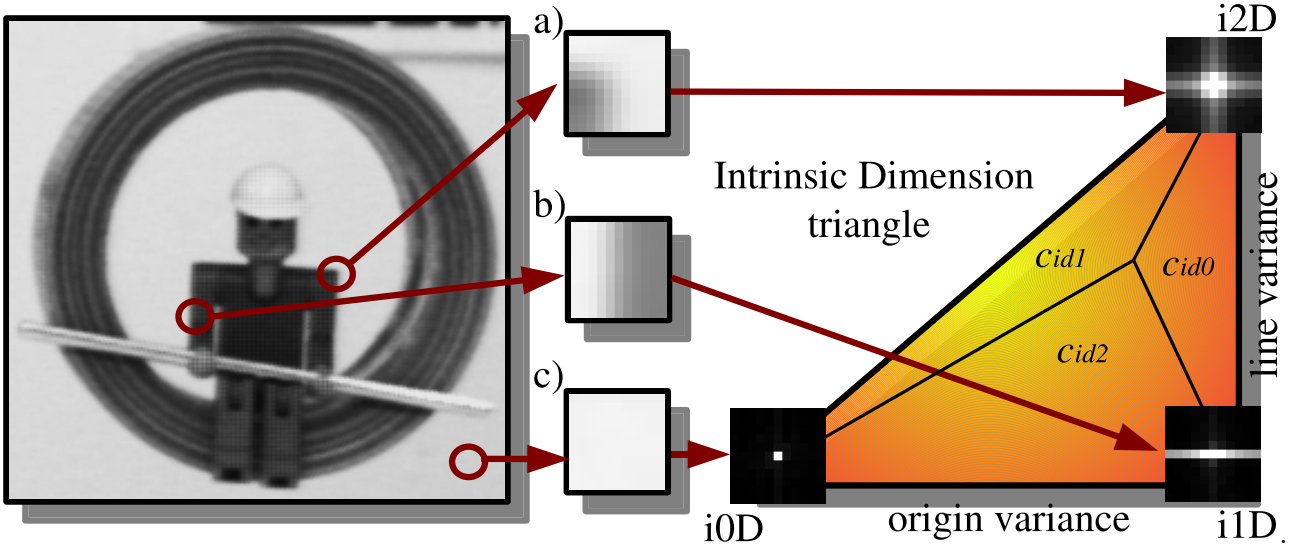


Figure 3: Illustration of the triangular topology of the intrinsic dimension — see [11]

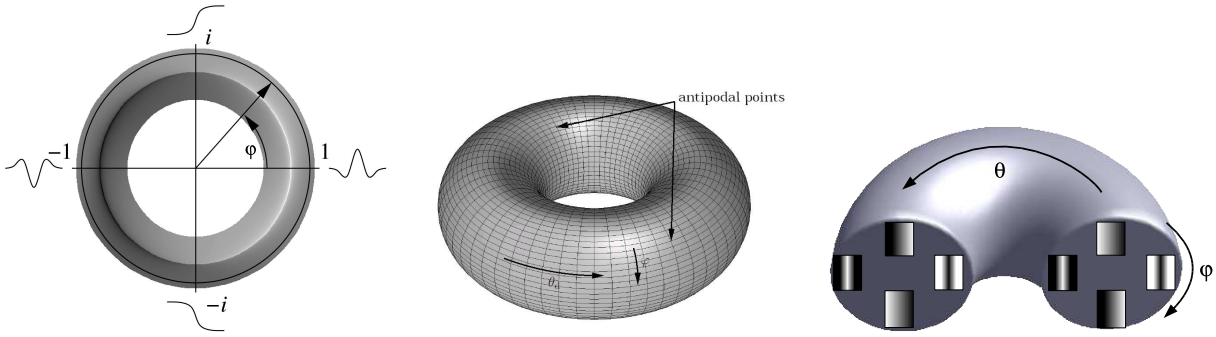
Hence, before applying concepts such as orientation or position, we need to classify image patches according to their junction–ness, edge–ness or homogeneous–ness. The intrinsic dimension (see, e.g., [51, 10]) has proved to be a suitable classifier in this context [11]. Ideal homogeneous image patches have an intrinsic dimension of zero (i0D), ideal edges are intrinsically 1–dimensional (i1D) while junctions and most textures have an intrinsic dimension of two (i2D). Going beyond common discrete classification [51, 21], we utilise a *continuous* concept [11, 12, 27] that allows for a formulation of reasonable confidences for the different image structure classes.

We classify image patches according to the dimension of the subspace that is occupied by the local spectral energy. When looking at the spectral representation of a local image patch (see figure 3), we see that the spectral energy of an intrinsically zero-dimensional signal is concentrated in the origin (figure 3a), whereas the energy of an intrinsically one-dimensional signal spans a line (figure 3b) and the energy of an intrinsically two-dimensional signal varies in more than one dimension (figure 3c).

It has been shown [27, 12] that the topological structure of the intrinsic dimensionality must be understood as a triangle that is spanned by two measures: origin variance and line variance. The origin variance describes the deviation of the energy from a concentration at the origin whereas the line variance describes the deviation from a line structure (see figure 3). We define the intrinsic dimension triangle such that each vertex corresponds to one ideal case of intrinsic dimension (homogeneous, linear or corner), and that its surface represents image patches that contains mixed aspects from these three ideal classes. It was shown in [11, 27, 12], that such a triangular interpretation allows for a *continuous formulation* of intrinsic dimensionality, parametrised by 3 confidences that are assigned to each of the mutually exclusive intrinsic dimension classes. For any image patch, the origin and line variances yield a point in this intrinsic dimension triangle (see figure 3d) and the confidence for this patch to belong to each of the three classes is computed using barycentric coordinates (see, e.g., [5]); namely, the confidence in a local patch to be of one of the classes (i0D, i1D or i2D) is the area of the sub–triangle defined by the origin and line variance of the patch, and by the ideal cases for the two other classes of intrinsic dimension — see figure 3.

Thus we compute for each pixel position $\boldsymbol{x}$ the three confidences $c_{id0}(\boldsymbol{x})$,$c_{id1}(\boldsymbol{x})$,$c_{id2}(\boldsymbol{x})$ that take values in $[0, 1]$ and add up to one — illustrated for different scales in the three bottom rows of figure 5. For details of the computation we refer to [11, 27, 12], and to [22, 23] for some applications of this concept.

The current version of our system focuses on intrinsically one dimensional signals and uses the triangular representation defined above to discard non–edge/non–line structures. There is some ongoing work on the integration of homogeneous (iD0) and corner structures (iD2) into this framework — see, [23, 50].



(a) phase (b) direction/phase space (c) orientation/phase space

Figure 4: a) The phase describes different intensity transitions, e.g., $\varphi = \pi$ encodes a dark line on bright background, $\varphi = -\pi/2$ encodes a bright/dark edge, $\varphi = 0$ encodes a bright line on a dark background and $\varphi = \pi/2$ encodes a dark/bright edge. The phase embed these distinct cases into a 2π-periodic continuum shown in (a). [Acknowledgement: Michael Felsberg] b) The torus topology of the orientation–phase space. The phase φ value is mapped on the cross section of the torus' tube whereas the orientation θ is maps to the revolution angle the torus. c) When direction is constrained to orientation (i.e. to the interval $[0, \pi)$) we get a half torus that is connected as indicated by the connecting strings.

2.2 Orientation and Phase

The extraction of a primitive starts with a rotation invariant quadrature filter that performs a *split of identity* of the signal [14]: it decomposes an intrinsically one-dimensional signal (as defined in the previous section) into local amplitude (see figure 5 top row), orientation (see figure 5 second row), and phase (symmetry, see figure 5 third row) information.[2]

The local amplitude is an indicator of the likelihood for the presence of an image structure. The orientation encodes the geometric information of the local signal while the phase can be used to differentiate between different image structures ignoring orientation differences. The phase for possible grey level structures forms a continuum between $[-\pi, \pi)$ and encodes the grey level transition of the local image patch across the edge (as defined by the orientation) in a compact way (as one parameter only), e.g., a pixel positioned on a bright line on a dark background has a phase of 0 whereas a pixel positioned on a bright/dark edge has a phase of $-\pi/2$ — see figure 4a and, e.g., [16, 26, 14]).

Note that phase is 2π-periodic and continuous such that a phase of $-\pi$ designate the same contrast transition as a phase of π.

Orientation θ (taking values in the the interval $[0, \pi)$) and phase φ are topologically organised on a half torus (see figure 4c), and if we extend the concept of orientation to that of a direction (therefore taking values in $[-\pi, \pi)$, see also [21]) then the topology of the direction/phase space becomes a complete torus (see figure 4b). On a local level the direction is not decidable[3] therefore we will use the half torus topology. This topology is crucial for the definition of suitable metrics for phase and orientation. For example, a black/white step edge ($\varphi = \pi/2$) with orientation θ should have small metrical distance to a white/black step edge ($\varphi = -\pi/2$) of orientation $\pi - \theta$ but large distance to a black/white step edge of orientation $\pi - \theta$. However, a white line on a black background with an orientation θ ($\varphi = 0$) should be have only a small distance to a white line on a black background with an orientation $\pi - \theta$ but a large one to any black line on a white background. Therefore the extremities of the half-torus are linked in a continuous manner as is shown in figure 4c. For a discussion of the orientation/phase metric we refer to [28, 40].

Figure 5 shows the filter responses in terms of the local amplitude $m(\boldsymbol{x})$, orientation $\theta(\boldsymbol{x})$ and phase

[2]Note that amplitude, orientation and phase can be analogously computed by Gabor wavelets or steerable filters and that our representation does not depend on the filter introduced in [14]. For a discussion of different approaches to define harmonic filters as well as their advantages and problems we refer to [43].

[3]Even taking the context into account there exists always two global solutions [16].

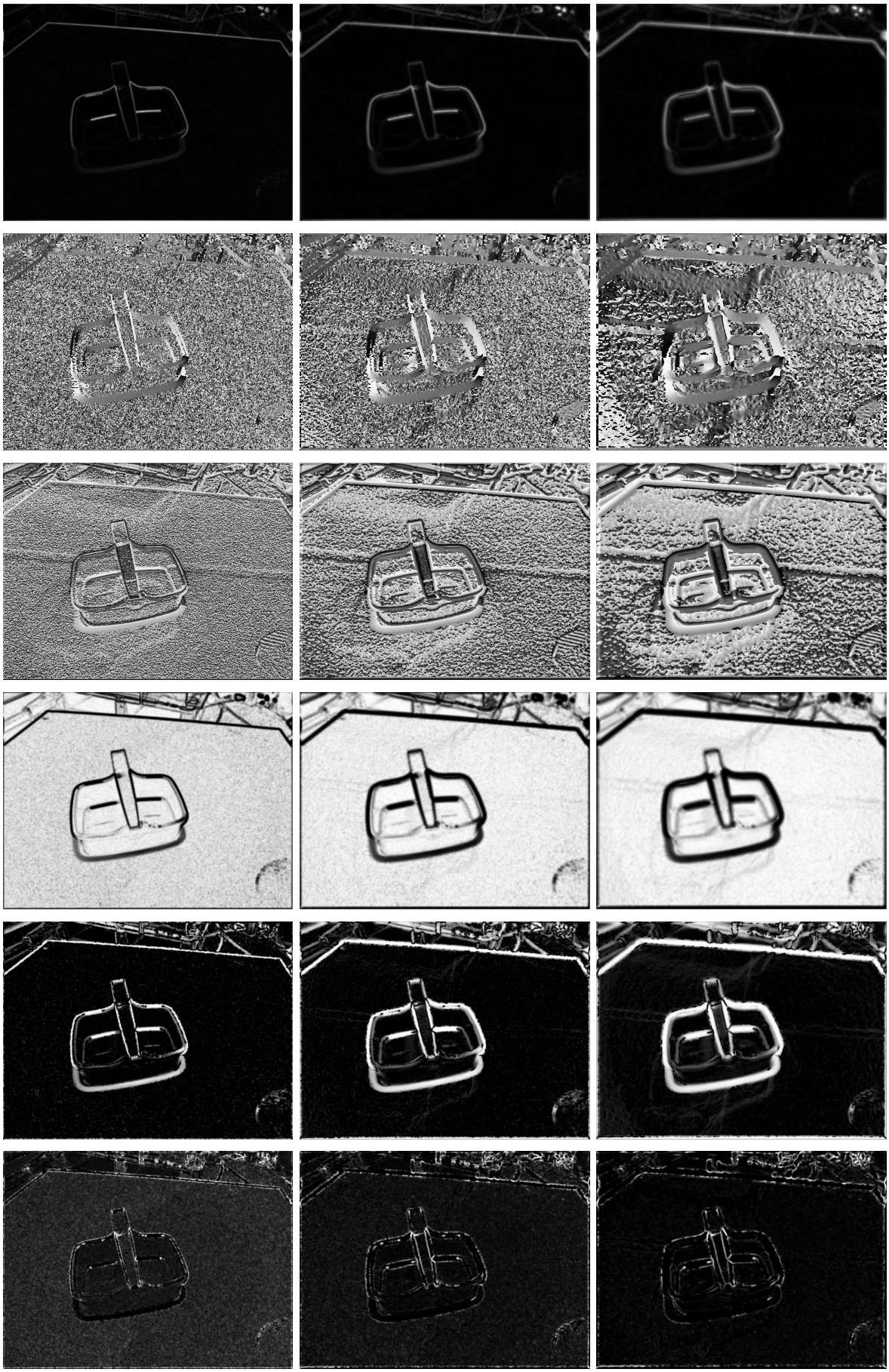


Figure 5: Illustration of the low–level processing for primitive extraction. Each column shows the filter response for a different peak frequency: respectively 0.110 (left), 0.055 (middle) and 0.027 (right). Each row show a response maps for, respectively from top to bottom, local amplitude,orientation, phase, intrinsically Zero–Dimensional (i0D), One–Dimensional (i1D) and Two–Dimensional (i2D) confidences. In all of those graphs white stands for high response and black for low ones.

$\varphi(\boldsymbol{x})$, alongside the resulting primitives, for three scales. The mathematical definition of the kernels and the split of identity is decried in appendix A.
The application of such a spherical quadrature filter for the processing of our Primitives has two main advantages:[4]

1) It allows us to utilise general advantages of the analytic signal (the aforementioned split of identity, see [16]). Hence, phase is an immediate output of the spherical quadrature filter processing and can directly be used as an attribute that describes the structural information of an oriented image structure (see figure 4A).

2) Compared to the use of a Gabor wavelet transform (see, e.g., [6]) we do not need to sample across different orientations but orientation is a direct output of the computation. Hence, we only need to apply 3 filter operations compared to, e.g., 16 for Gabor wavelets (see, e.g., [33]).

We compute filter responses for three different scales (the three scales used in the present work are described in appendix A).[5]

2.3 Optic Flow and Colour

Besides orientation, phase and the intrinsic dimensionality confidences, colout and the local optic flow vector is also associated to the primitive description vector.
In [22], we compared the performance of different optic flow algorithms depending on the intrinsic dimensionality, i.e., the effect of the aperture problem and the quality on low contrast structures. It appeared that different optic flow algorithms might be optimal in different contexts. In our system we primarily use the Nagel–Enkelmann algorithm [38] since it gives stable estimates of the normal flow at i1D structures. We denote the optic flow computed at a position $\boldsymbol{x}$ by $\boldsymbol{f}(\boldsymbol{x})$.
Colour is not processed by (non–)linear filtering operations but sampled (i) on each side of a step edge, or (ii) on each side of a line and on the line itself, depending if the phase describes a step edge or line structure.

3 Condensation Scheme

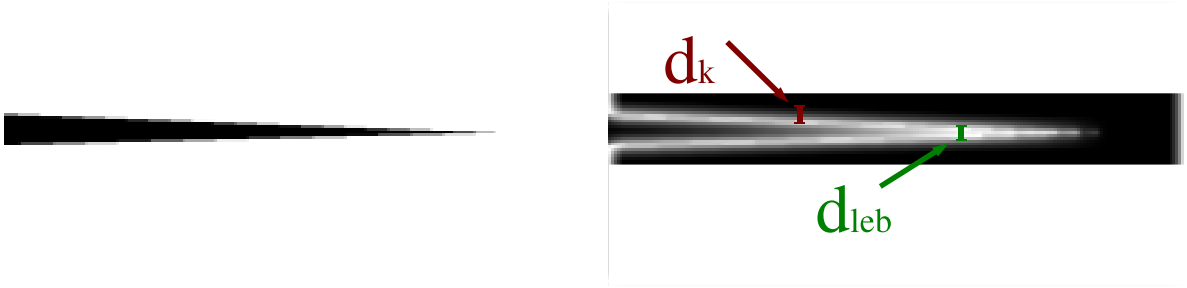
Based on the pixel–wise processing described in section 2, we now want to extract a condensed interpretation of a local image patch by selecting a sparse set of points to which visual modalities become associated. An important aspect of the condensation scheme is that all main parameters can be derived from one property of the basic filter operations called *line–edge bifurcation distance*. This value expresses the minimal distance between two edges for them to be represented by two distinct primitives. Below this distance, one single line primitive will be extracted. In 6(a) shows a narrow triangle for which two edges get closer until the vertex. Vertical sections of the local local amplitude (b) close to the vertex features only one maximum, whereas it splits into two distinct maxima further on, where the triangle is broader.

Definition. *The line–edge bifurcation distance d_{leb} for a given scale is the minimal distance between two edges for them to produces two distinct maxima.*

Using the above definition we propose a condensation procedure in three steps:

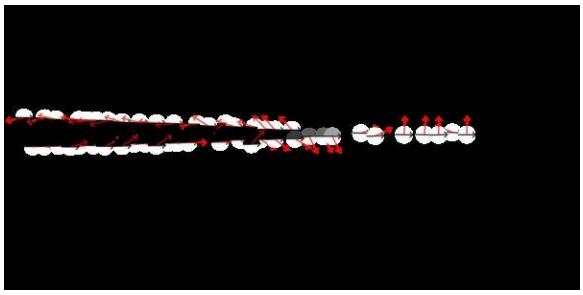
[4]Note that there are also some problems involved with filters realising the monogenic signal we are suing. These are discussed in [43]. First, it turned out that for the monogenic signal it is more difficult to construct filter which allow for stable orientation and phase estimates at high frequencies (compared to, e.g., Gabor wavelets) Second, in the monogenic filter approach there is only one orientation estimate and one phase (in connection to the one orientation) estimate. However, for intrinsically two dimensional signals such as corners and most textures more parameters are needed to represent the local structure (e.g., most textures are characterised by multiple orientations at different frequencies). Third, estimates for, e.g., optic flow can profit from averaging processes over estimates over different orientations. However, in the context of intrinsically one dimensional structures the monogenic signal allows for a good representation.

[5]Note that for step edges, we can expect high amplitudes over different frequency levels, while line structures might become represented at a high frequency level as two step-edges and on a lower frequency level as a line (see section 3).

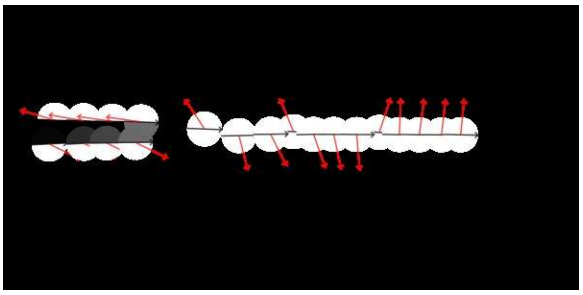


(a) original image

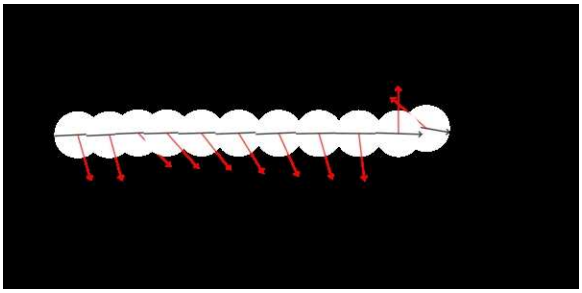
(b) local amplitude



(c) peak frequency 0.110



(d) peak frequency 0.055



(e) peak frequency 0.027

Figure 6: Definition of the elimination parameters d_{leb} and d_k. See text for an explanation.

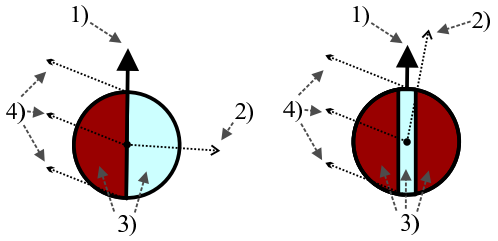
Sampling: The positions of features are computed with sub–pixel accuracy, according to the local intrinsic structure (section 3.1).

Elimination: Positions that are too close to each other (and therefore would lead to redundant descriptors) become deleted (section 3.2).

Local Interpretation: Semantic attributes become associated to the computed positions. (section 3.3).
Figure 6 (c), (d) and (e) shows the primitives extracted after condensation for the three scales used in the present paper — for peak frequencies of 0.11, 0.055 and 0.027, respectively.

3.1 Sampling

In section 2.1 it was discussed that the concept of position is different for different type of image structures as defined by the three classes of intrinsic dimensionality.
The coding of intrinsic dimension by three values $c_{i0D}, c_{i1D}, c_{i2D}$ allows us to select the most likely structure for this patch, and thence to define an appropriate (according to its intrinsic dimension interpretation) position candidate. However, if we do not want to make a decision about the type of local image structure at such an early stage we can also code the three different candidates according to their intrinsic dimension class (see figure 8b). These two approaches are implemented by two different modes of the condensation algorithm with different advantages and disadvantages (see below).



(a) An edge primitive (b) A line primitive

Figure 7: Illustration of the symbolic representation of a primitive for a i1D interpretation, for a) a bright-to-dark step–edge (phase $\varphi \neq 0$) and b) a bright line on dark background (phase $\varphi \neq \frac{\pi}{2}$. 1) represents the orientation of the primitive, 2) the phase, 3) the colour and 4) the optic flow.

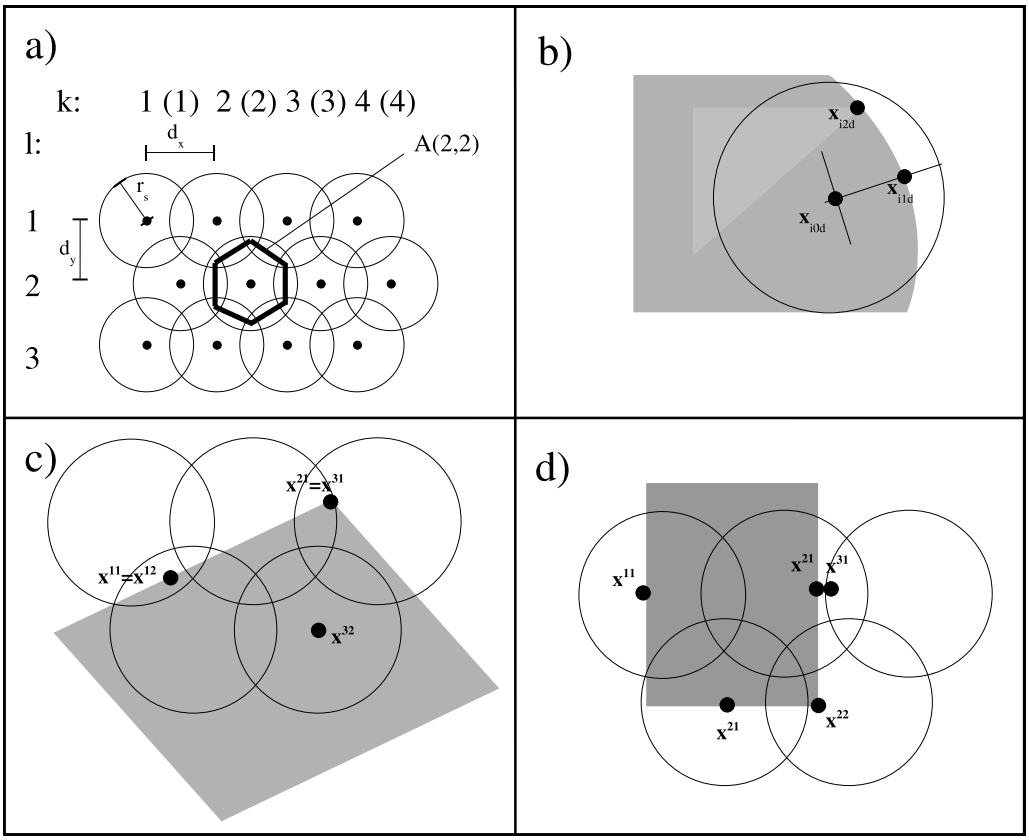


Figure 8: a) Hexagonal Sampling. b) Three possible hypotheses for positions according to the three different intrinsic dimensions. c) Because of the overlap in the hexagonal sampling the same position can be found in areas with different index. For these redundant structures one sample needs to be deleted. d) Since the local amplitude can still be high for pixels with a certain distance from high contrast structure it might be that a position is found that is actually not on the edge structure. These points represent redundant structures since they are already represented more accurately (in terms of position) by other primitives. These hypotheses need also to be deleted.

Figure 9: Top row: positions associated to the primitives assuming different intrinsic dimensionality (from left to right, i0D, i1D and i2D). Middle row: Primitives in each of those cases (from left to right, i0D, i1D and i2D). Bottom row, left: positions using the interpretation given by the intrinsic dimension with the highest confidence; middle: primitives extracted at those locations; and right: primitives at non-i0D locations.

Peak frequency	f_p	0.1103	0.0551	0.0275
Wavelength	f_p	9.06	18.12	36.25
Number of tabs	n_t	11	23	33
Line/edge bifurcation	d_{leb}	3	6	7.5
Hex. grid spacing in x	$d_x = 0.85 d_{leb}$	2.55	5.1	6.37
Hex. grid spacing in y	$d_y = \sqrt{3}/2 d_x$	2.21	4.42	5.52
2^{nd} elimination param.	$d_k = 2.2 d_{leb}$	6.6	13.2	16.5
Condensation rate	d_{co}	85%	94%	97%

Table 1: Frequency–dependent parameters

To get candidates for our primitives, we first perform an hexagonal sampling (see figure 8a) of the image into overlapping areas $A^{(k,l)}$ with radius r_s, with k, l coding the hexagonal grid points. Hexagonal sampling has a number of advantages discussed for example in [47, 37].[6] In the context of this paper, the most important difference to a rectangular sampling is that in case of hexagonal tiles the distance between the midpoints of neighbour tiles is uniform whereas in a rectangular grid diagonal neighbours are are $\sqrt{2}$ times further than horizontal or vertical neighbours. Since we want to extract symbolic descriptors for each tile, the hexagonal sampling allows for a more evenly distributed symbolic description and also reflects more closely the isotropic structure of the original image filters.

The sampling distance depends on the *line/edge bifurcation distance* and thereby on the peak frequency for the scale being used (note that it is also related to the spatial size, and the minimal number of tabs n_t, needed to represent the filter, for a detailed discussion see [43]). The parameters d_x and $d_y = \frac{\sqrt{3}}{2} d_x$ determine the spatial distance in x and y between the centre $A_c^{(k,l)}$ of the tile $A^{(k,l)}$ and the centres of the neighbour tiles.[7] For a description of the mathematics of the hexagonal sampling we refer to, e.g., [37].

The sampling distance d_x is related to the line/edge bifurcation distance d_{leb} that depends of the peak frequency f_p and the band-width B of the filter applied.

In appendix A we describe the derivation of the kernels of the monogenic signal which bandpass characteristics are controlled by the two parameters s_1 and s_2. The peak frequency is computed by

$$f_p = \frac{1}{2\pi(s_2 - s_1)} ln(\frac{s_2}{s_1}) \tag{1}$$

Since in our case we have $s_2 = 2s_1$ this becomes

$$f_p = \frac{1}{2\pi s_1} ln(2) \tag{2}$$

with s_1 set to 1, 2, and 4 covering the frequency domain in a reasonable way (see figure 19).

It turned out that a reasonable estimate for d_{leb} is

$$d_{leb} = \frac{1}{3 f_p} \tag{3}$$

hence we set

$$d_x = round(d_{leb}) + 1 \tag{4}$$

being the smallest possible sampling distance within which structures based on the amplitude information can be resolved. All frequency depended parameters are shown in table 1:

We search on disk around each $A_c^{(k,l)}$ for candidate positions of primitives. The radius r_s of this disk is chosen such that each point of the image is covered by at least one of the disks. In a hexagonal

[6] For example Mersereau [36] showed that hexagonal sampling is optimal for certain band limited signals.

[7] Note that the odd rows have an onset of $d_x/2$

grid, the maximum distance to the border of a tile is $\frac{2}{\sqrt{3}}d_x$ hence we set

$$r_s = round(\frac{2}{\sqrt{3}}d_x) + 1 \tag{5}$$

We then look for optimal structure dependent positions inside each tile, distinguishing between the three intrinsic dimension classes:

i0D **Homogeneous image patches:** At homogeneous image patches the position can not be defined by properties of the local signal since it is constant. Therefore, the position $\boldsymbol{x}_{id0}^{(k,l)}$ of a Primitive representing an image patch $A^{(k,l)}$ is defined by the equidistant sampling (see figure 2a):

$$\boldsymbol{x}_{id0}^{(k,l)} = A_c^{(k,l)}$$

i1D **Lines and edges:** For a line or edge, the position $\boldsymbol{x}_{id1}^{(k,l)}$ can be defined through energy maxima that are organised as a one–dimensional manifold. Therefore, an equidistant sampling along these energy maxima is appropriate (see figure 2b). For this, we look for the energy maximum along a line orthogonal to the orientation at $A_c^{(k,l)}$ which is within the area $A^{(k,l)}$.

$$\boldsymbol{x}_{id1}^{(k,l)} = \max_{\mathbf{x} \in g^{(k,l)}} m(\boldsymbol{x})$$

where $g^{(k,l)}$ is a local line going through $A_c^{(k,l)}$ with orientation perpendicular to $\theta(A_c^{(k,l)})$.

i2D **Junction–like structures:** For a junction the position $\boldsymbol{x}_{id2}^{(k,l)}$ can be defined unambiguously as the maximum of the i2D confidence in a local region (see figure 2c and [13]):

$$\boldsymbol{x}_{id2}^{(k,l)} = \max_{A^{(k,l)}} \{c_{id1}(\boldsymbol{x})\}.$$

Our system runs in two modes. In the first mode, hereafter named *complete mode*, all three hypotheses are conserved (see figure 8b), however the position corresponding to the maximum of three confidences $c_{i0D}, c_{i1D}, c_{i2D}$ is called the *external position* $\boldsymbol{x}^{(k,l)}$ and it is used in the following process of reduction of redundant descriptors to compete with candidates computed in other tiles of the hexagonal grid. In the second mode, named *contour mode*, we only look at intrinsically one-dimensional signals, i.e., we do the positioning according to figure 2b. The first mode allows for a complete representation of the signal by also taking into account i0D and i2D structures. However, the symbolic representation as well as the 3D reconstruction of i0D and i2D signals differ and are ongoing research topics (see, e.g. [23, 50]). In the second mode, the symbolic representation of the primitives, their 3D reconstruction (see section 4) as well as important structural relations between primitives such as co-colority, symmetry and co-planarity are defined (see section 5.1).
All positions are computed with sub-pixel accuracy using the formula :

$$\begin{array}{rcl} \tilde{x}_0 & = & \frac{1}{s_g} \sum_{i=-w_s}^{w_s} \sum_{j=-w_s}^{w_s} m(x_0+i, y_0+j)(x_0+i) \\ \tilde{y}_0 & = & \frac{1}{s_g} \sum_{i=-w_s}^{w_s} \sum_{j=-w_s}^{w_s} m(x_0+i, y_0+j)(y_0+i) \end{array} \tag{6}$$

with $m(x, y)$ being the local amplitude at pixel position (x, y) and

$$s_g = \frac{1}{\sum_{i=-w_s}^{w_s} \sum_{j=-w_s}^{w_s} m(x_0+i, y_0+j)} \tag{7}$$

where w_s is set to $w_s = d_{leb}$ In section 3.3 the modalities phase and orientation of the extracted features are computed at the sub-pixel accuracy position by bi-linear interpolation.
Figure 9 shows the positions found for different intrinsic dimensions and also the external positions.

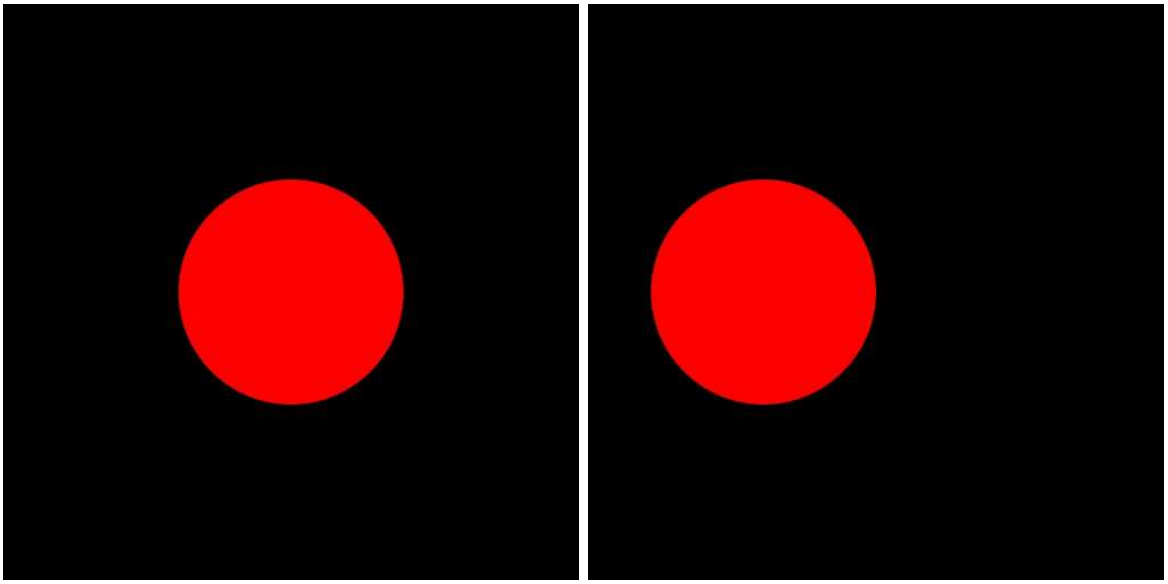
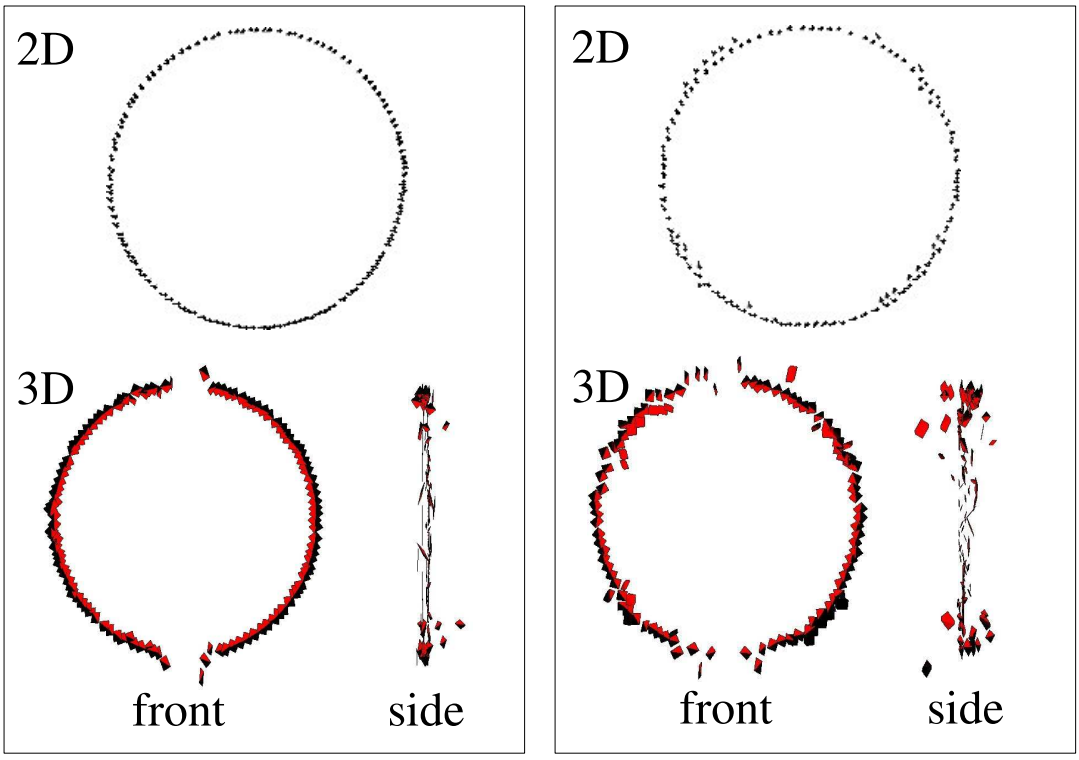


Figure 10: Artificial sequence used to evaluate the accuracy of primitive extraction (see figure 11).

Figure 14 shows the primitives extracted from a simple indoor scene (a). The primitives are extracted with a an origin variance > 0.3 and a line variance < 0.3 are shown for the three scales considered in this work: namely for peak frequencies of 0.110 (b), 0.055 (c), and 0.027 (d). Different scales highlight different structures in the scene.
In figure 10 an artificial sequence featuring a red circle on black background is shown. We evaluated the accuracy of the primitive extraction on this scene, and the results are recorded in figure 11. The top images compare the primitives extracted with (a) and without (b) the sub–pixel localisation of the primitives. Note that the sub–pixel localisation requires a symbolic interpretation of the primitive and that therefore we only considered i1D primitives. Effectively we only considered primitives with an origin variance larger than 0.3 and a line variance lower than 0.3. The upper graphs in (a) and (b) show the 2D primitives extracted and whereas the bottom ones show the 3D–primitives reconstructed using stereopsis.

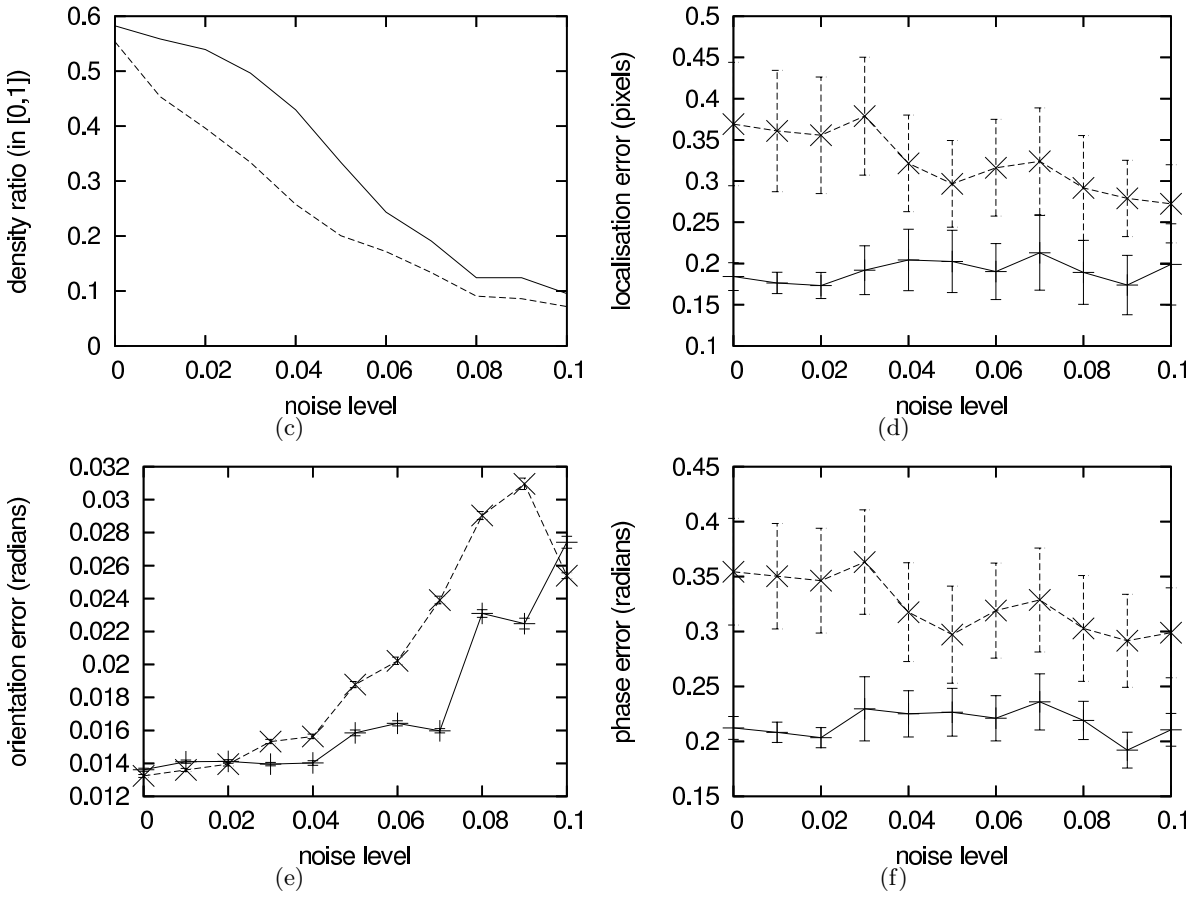
3.2 Elimination of redundant descriptors

Since the areas $A^{(k,l)}$ are overlapping, the process described above can lead to identical positions found in neighbouring areas (see figure 8c, $\boldsymbol{x}^{(2,1)} = \boldsymbol{x}^{(3,1)}$, $\boldsymbol{x}^{(1,1)} = \boldsymbol{x}^{(1,2)}$). And since the applied filters are extended in space it can also lead to positions with close spacing describing essentially the same structure (see figure 8d, $\boldsymbol{x}^{(2,1)}$ and $\boldsymbol{x}^{(3,1)}$.
In the second step described now, these redundant positions become eliminated. In this elimination process we face the following difficulty: On the one side, we do not want to eliminate 'independent structures' that are close to each other. For example, in the triangle in figure 6 two edges converge. At some point, these edges become interpreted as a line and the position should be on this line and the phase should become 0 or $\pm\pi$. Until then, the triangle should be represented by two edges with phase $\pm\pi$. Hence, the elimination process should not eliminate these 'independent' edges although they can be rather close to each other. The limit of separability is the line/edge bifurcation distance d_{leb} defined above. On the other side, since our kernels have an extension (expressed in the number of tabs n_t used to approximate the spatial filter) that is larger than d_{leb} there will still be a significant amplitude at pixel distances larger than d_{leb} (see figure 6).
As a consequence, eliminating candidates with distance smaller than d_{leb} would preserve all 'independent' edge structures but would also preserve a lot of redundant structures. However, eliminating candidates with distance smaller than n_t would eliminate all redundant but also the 'independent' structures.



(a) with sub–pixel localisation

(b) without sub–pixel localisation



(c) (d) (e) (f)

Figure 11: a) and b): 2D– and 3D–primitives extracted in the scenario illustrated in figure 10, respectively with and without sub–pixel localisation. c), d), e) and f) report the density and accuracy in localisation, orientation and phase of the primitives, wherein the solid line show the accuracy with sub–pixel localisation and the dashed line without. The error bars in d), e) and f) show the variance.

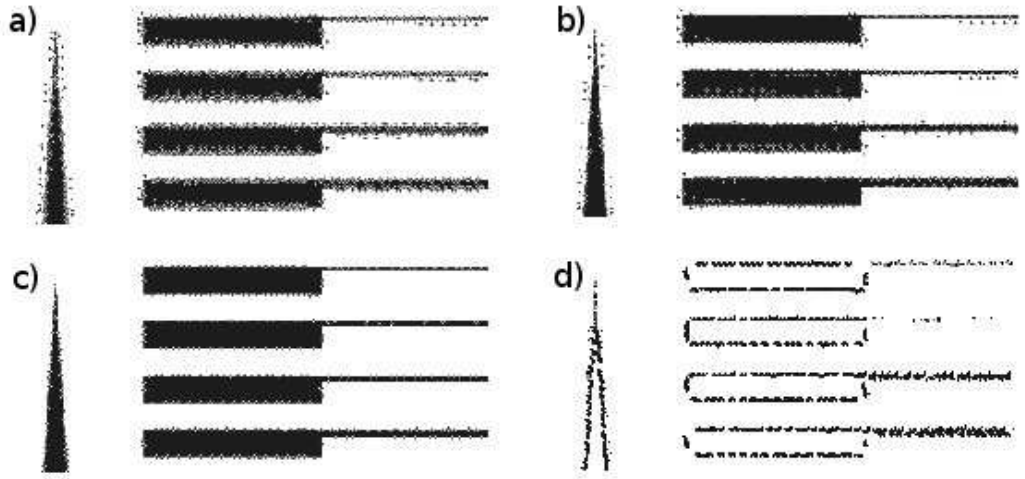


Figure 12: Three stages of the elimination process and the final primitive representation.

We tackle this problem by a two stage elimination process described in section 3.2.1 and 3.2.2.

3.2.1 Elimination based on the line/edge bifurcation distance

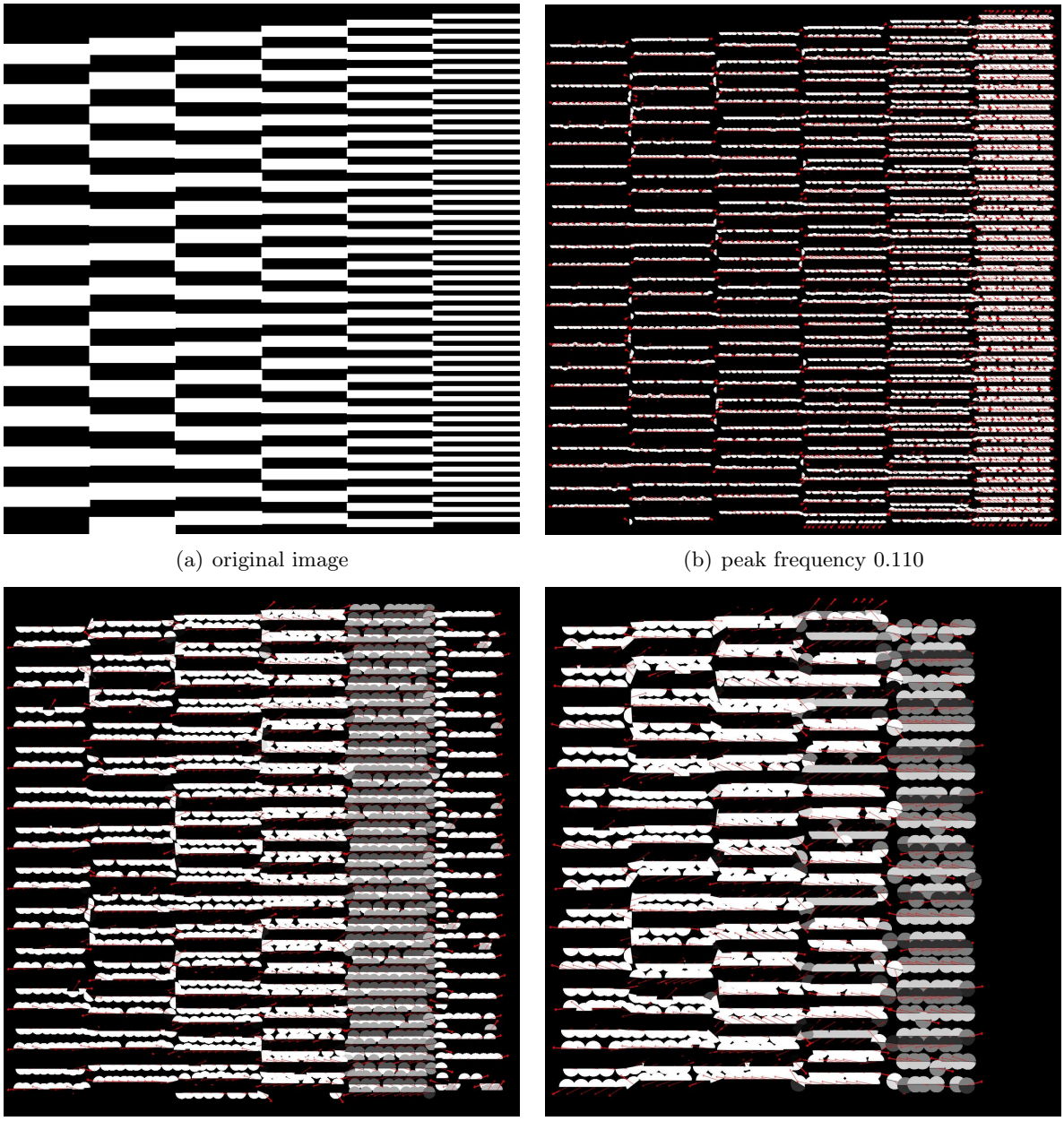
First, all candidates $\boldsymbol{x}^{(k,l)}$ become ordered according to the associated amplitude $m(\boldsymbol{x}^{(k,l)})$. Starting with candidates with the highest local amplitude we delete all other candidates $\boldsymbol{x}^{(k',l')}$ with a distance $d(\boldsymbol{x}^{(k,l)}), \boldsymbol{x}^{(k',l')}) = ||\boldsymbol{x}^{(k,l)}) - \boldsymbol{x}^{(k',l')}||$ smaller than d_{leb}.[8] Since we order the candidates according to the local amplitude, the candidate corresponding to a 'stronger' structure suppress candidates with weaker structure. Thereby all non–distinct edges (according to the line edge bifurcation distance) become deleted but redundant edges are still being preserved. In figure 12 upper–left, we see that many spurious candidates remain after the first elimination process that are caused by edges with distance smaller then d_k.

3.2.2 Elimination based on the kernel size

In the second step, again starting with the candidates with highest local amplitude, all remaining candidates become tested according to a distance d_k. d_k expresses the distance to which a structure can essentially effect pixels in the vicinity and is set to $d_k =$.
For a pair of intrinsically two dimensional structures it is sufficient to have distance smaller than d_{leb} since they naturally represent maxima in the amplitude representation [13]. If an intrinsically one-dimensional structure is involved there will be a slant in the local amplitude surface at the 'dependent' structure having its maximum at the edge/line structure decreasing with distance from the edge (see figure 6). This slant can be checked for: For each pair of candidates found with distance smaller d_k a test is made whether it represents an 'independent' structure. The criterion for independence we are using is whether the structure is a maximum on the line orthogonal to the local orientation. For each remaining candidate the amplitude is compared to the amplitude at pixels at a distance $d_{co} =?$ at both sides of the edge indicated by the local orientation.[9] If that is the case the candidate with lower local amplitude is discarded.
Thereby the remaining spurious candidates become eliminated. Figure 13 shows the primitives extracted for an artificial test image, for different scales. The figure in (a) shows vertically alternating black/white step–edges, getting narrower to the right of the figure. The primitives extracted at the three scales, for peak frequencies of 0.110, 0.055 and 0.027, are shown in (b), (c) and (d), respectively. The different effect of the double elimination process at different scales can be seen in this figure. For

[8] Note that for the quality of the process it is important that all positions are computed with sub-pixel accuracy already at this stage.

[9] Note that the criterion 'local maxima' that is applicable for i2D structures can not be applied since edge like structures form a ridge in the local amplitude surface (see figure 6).



(a) original image

(b) peak frequency 0.110

(c) peak frequency 0.055

(d) peak frequency 0.027

Figure 13: Dense sampling.

example if all of the narrower step edges to the right of the figure are distinctly extracted in (b), only one of the two is extracted in (c), whereas in (d) the same edges become intrinsically two–dimensional and are not extracted anymore.

3.3 Association of Visual Attributes and Confidences

Based on the found positions $\boldsymbol{x}^i$ we can associate visual attributes. The attributes orientation θ, phase φ, and optic flow $\boldsymbol{f}$ are computed pixel-wise using filter processes of spatial extend d_k[10]. Therefore, we associate the orientation, phase, as well as the optic flow according to the found positions $\boldsymbol{x}^i$.
Since, positions are computed with sub–pixel accuracy we can also interpolate the orientation, phase and optic flow value by bi–linear interpolation []. Let $\tilde{x}_0$ and $\tilde{y}_0$ be the positions computed with sub-pixel accuracy (see section 3.1). Let δ_x and δ_y be the distance to the discrete lower pixels x_l and y_l (and $x_h = x_0 + 1$ and $y_h = y_0 + 1$, then the bi-linear interpolation computation leads to the formula:

$$\begin{aligned}\tilde{\theta}(\tilde{x}) \quad = \quad & \hat{\theta}(x_l, y_l)(1-\delta_x)(1-\delta_y) + \hat{\theta}(x_l, y_h)(1-\delta_x) * \delta_y \\ & \hat{\theta}(x_h, y_l)\delta_x(1-\delta_y) + \hat{\theta}(x_h, y_h)\delta_x\delta_y\end{aligned}$$

Note that for the interpolation of orientation and phase the specific topology of the orientation phase space needs too be taken into account. Hence $\hat{\theta}$ is transformed such that the distance between all pairs of the set $\hat{\theta}(x_l, y_l), \hat{\theta}(x_l, y_h), \hat{\theta}(x_h, y_l), \hat{\theta}(x_h, y_h)$ is smaller than $\frac{\pi}{2}$ and $\hat{\theta}(\tilde{x})$ is in $[0, \pi)$. Phase is computed analogously.[11]
For the test picture shown in figure 10 we get a localisation error in the area of 0.1 pixel (i.e., improvement of a factor 10). Bi-linear interpolation of orientation and phase based on the the sub-pixel accuracy positioning leads also to improvements of a factor 2 and 6 respectively (on the highest frequency level). The effect on reconstruction is also demonstrated in figure 11.
Also colour information is available for each pixel position. However, especially for i0D and i1D signals the representation of colour is highly redundant. For a step-edge like structure it is natural to distinguish between the colour on the left and right side of the edge $(\boldsymbol{c}_l, \boldsymbol{c}_r)$ while for a line structure also the colour of a middle strip $\boldsymbol{c}_m$ should be coded (see figure 6c–e and 7).
As discussed in section 2.2 by the phase we can distinguish these two cases. For an homogeneous image patch (i0D), colour pixels can even be subsumed into one colour attribute.
Finally, we have a parametric description of a local area that we call a primitive. For a step edge we get

$$\pi_i = (\boldsymbol{x}_i, \theta(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_i), (\boldsymbol{c}_l(\boldsymbol{x}_i), \boldsymbol{c}_r(\boldsymbol{x}_i)), \boldsymbol{f}(\boldsymbol{x}_i))$$

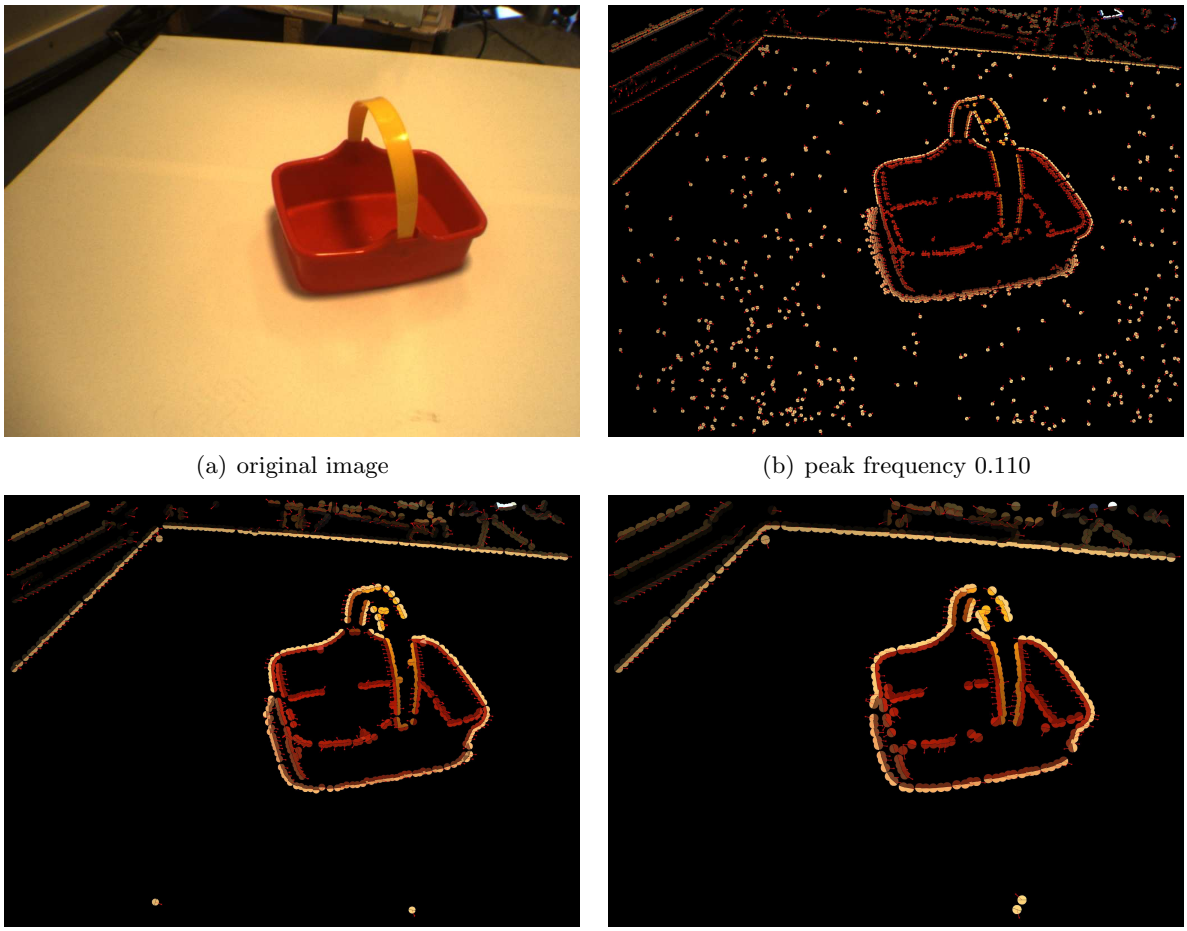
while for a line we get

$$\pi_i = (\boldsymbol{x}_i, \theta(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_i), (\boldsymbol{c}_l(\boldsymbol{x}_i), \boldsymbol{c}_m(\boldsymbol{x}_i), \boldsymbol{c}_r(\boldsymbol{x}_i)), \boldsymbol{f}(\boldsymbol{x}_i))\,.$$

The parameters of the primitives have a clear semantic and are a condensed representation of the local image patch. Condensation can be computed by the ratio of the number of bit needed to store a local image patch a primitive stands for. For the highest frequency, such a primitive represents a local image patch of a radius of appr. 3 pixels (i.e., $\pi \cdot 3^2 \cdot 3 \approx 85$ values). The primitive has a dimension of 10 for an edge like structure and 13 for a line–like structure(not counting the optic flow which indicates temporal information). That means that a primitive for the highest frequency level only requires maximal the $\frac{13}{85} = 0.15$ amount of bytes compared to the original image information leading to a condensation rate $d_{co} \approx 85\%$. Analogously, we get a condensation rate of $\approx 94\%$ and $\approx 97.\%$ for the other two frequency levels. Note that after computing the 3D primitives (see section 4) the condensation rate increases again significantly.

[10] Phase and orientation are output of the spherical quadrature filters while the area the optic flow estimation is based on can be determined in different flow algorithms in different ways.

[11]
$$\varphi(\tilde{x}) = \text{to be made}$$



(a) original image

(b) peak frequency 0.110

(c) peak frequency 0.055

(d) peak frequency 0.027

Figure 14: 2D–primitives extracted for different peak frequencies

Table 1 shows all parameters included in the primitive extraction. Note that these parameters are either naturally derived from the line edge bifurcation distance (d_{leb}) or are non-critical (w_s) or are based on decisions involving a trade off between computational complexity and precision (d_k).

4 Computation of 3D-Primitives

So far we have described multi-modal *image* descriptors that code *2D* information. However, these descriptors describe visual events occurring at a certain 3D position in space. This depth information is of essential use for higher level processes because of two reasons. First, human and robots act in a 3D world where depth information gives valuable indication where actions such as moving or grasping are possible. Second, since many structural dependencies of visual events (e.g., rigid body motion) are working on 3D structures the association of 3D information is essential for the formalisation of the disambiguation processes (see [41]).
In the following, we describe an extension of the image primitives to spatial primitives. In these spatial primitives, the semantic information coded in the image primitives is transferred into the 3D domain. Therefore we need to come to good interpretations of image information as 3D events.
Assuming the correspondences between primitives in two images are known (for how this is done, see [41]) we are able to extract spatial primitives as described in section 4.1 (see also figure 16).

4.1 Constructing Spatial multi–modal Primitives

Given a pair of corresponding points between the left and right image, a meaningful 3D interpretation of this stereo–pair is a 3D point. Contours, however, hold a 2D orientation, and therefore 3D–primitives need to encode the reconstructed 3D orientation $\boldsymbol{\Theta}$ beside the 3D position $\boldsymbol{X}$; this orientation is computed as the intersection of two planes in space, each defined by the optical centre of one camera and the line in the image plane described by the image primitive's position and orientation — see figure 15. The intersection of these two planes in space is a 3D line that provides us with the orientation of the 3D primitive. In [48] it was shown that using line correspondences for the reconstruction of 3D orientation was generally more accurate than points correspondences.
Phase and colour are reconstructed in space as the mean value between the two corresponding image primitives.

$$\Phi = \frac{\varphi^L + \varphi^R}{2} \tag{8}$$

$$\boldsymbol{C} = \frac{\boldsymbol{c}^L + \boldsymbol{c}^R}{2} \tag{9}$$

Moreover these two modalities encode surface information (respectively contrast and colour transition across an edge) thus we need to define a 3D surface patch onto which they apply. Unfortunately it is not possible to reconstruct the exact surface from local information: for a pure i1D signal the surface on one side does not allows to find the additional correspondence that would be required for the reconstruction of a 3D surface. Moreover, in case of a depth discontinuity the colour information might come from a 3D position that is completely independent from the 3D orientation information (i.e. the background).
We propose to define as *a priori* 3D surface the plane that is most stable under small viewpoint variation (see figure 15). This surface is computed using the 3D orientation of the primitive and an additional vector Γ that is defined as follows:

$$\Gamma = \boldsymbol{\Theta} \times V_{pov} \tag{10}$$

such that the surface is normal to V_{pov}, and V_{pov} is defined as follows

$$V_{pov} = \frac{1}{2}\left(\overrightarrow{C_L\boldsymbol{X}} + \overrightarrow{C_R\boldsymbol{X}}\right) \tag{11}$$

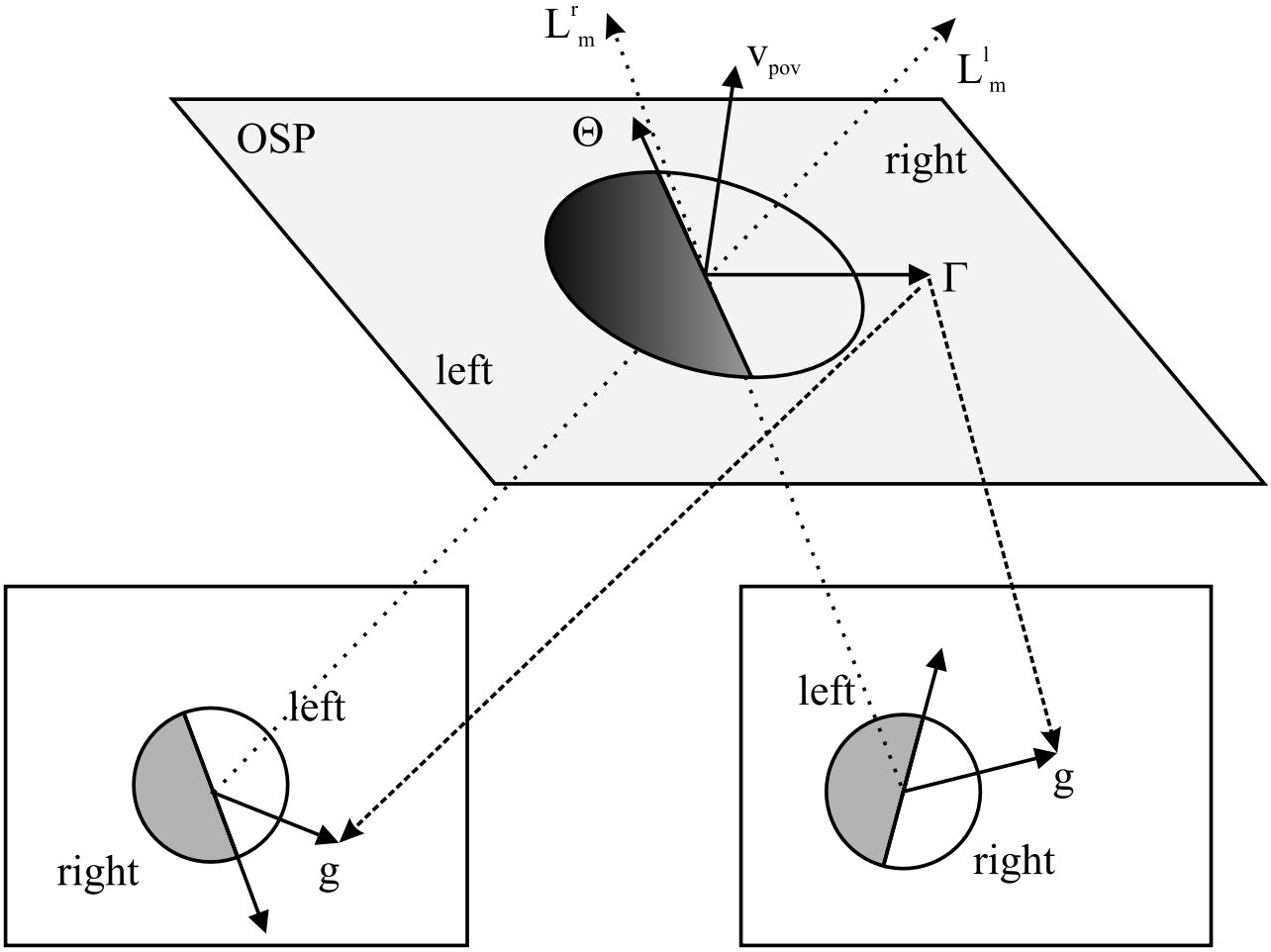


Figure 15: Illustration of the reconstruction of the 3D orientation.

where $\overrightarrow{C_L\boldsymbol{X}}$ and $\overrightarrow{C_R\boldsymbol{X}}$ are the two optical rays joining the location of the primitive $\boldsymbol{X}$ with the optical centre of the left (C_L) and right (C_R) cameras. The vector Γ also identifies each side of the 3D line, which is critical for modalities like colour and phase that describe the modality transition across the contour.
We end up with a set of spatial primitive $\Pi^{(i,j)}$ each having the parametric description

$$\Pi^{(i,j)} = (\boldsymbol{X}, \boldsymbol{\Theta}, \Phi, (\boldsymbol{C}_l, \boldsymbol{C}_m, \boldsymbol{C}_r)) \tag{12}$$

The j-index represents the alternative 3D entities generated from different correspondences in the right image to the i-th primitive in the left image. Since a final decision can usually not be made with high reliability solely based on local information, multiple hypotheses are kept at this stage. In the following section we will describe different approaches to overcome this ambiguity.
In figure 11 (a) and (b), bottom, the 3D primitives reconstructed with (a) and without (b) sub–pixel localisation are shown from front and side view. The side view offer a better vision of the quality of the depth estimation from stereopsis.[12] It is visible in these images that the sub–pixel localisation of the primitives described in section 3.1 allows for a notably better 3D–reconstruction.
In figure 16 the 3D–primitives reconstructed in an indoor scene are shown. Figures (a) and (b) show the stereo pair of images used, (c) (resp (d)) shows the 2D–primitives extracted with (resp. without) sub–pixel accuracy, and the subsequently reconstructed 3D–primitives are shown in (e) (resp. (f)).

5 Applications

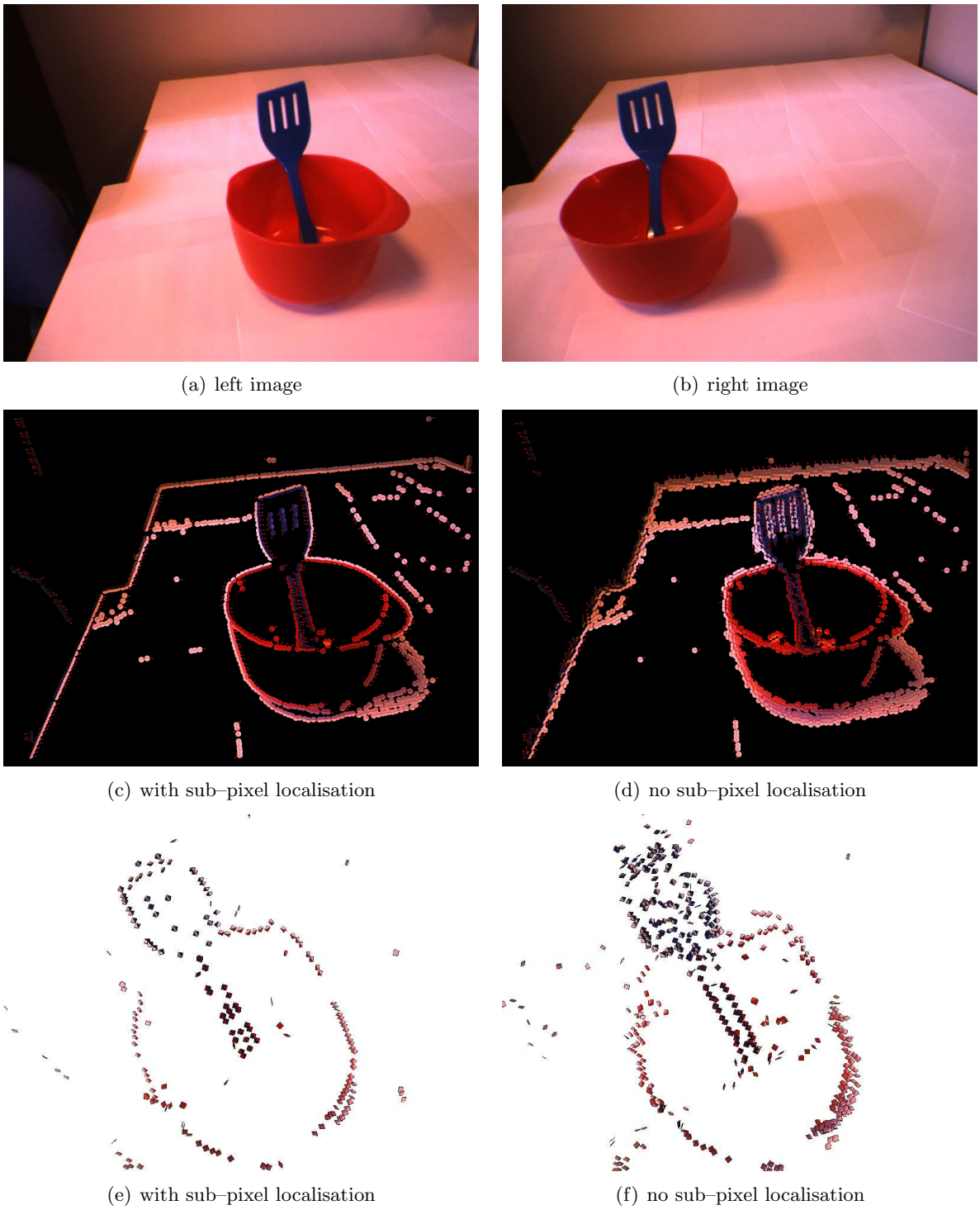
The primitive representation introduced in this paper has been applied in various contexts (briefly described in subsection 5.2 to 5.5) and has been part of three different European projects [8, 1, 18] in the area of Cognitive Vision and visual based robotics. The primitives described so far are condensed localised descriptors with clear semantics, and by this, symbolic descriptors of a local image patch. Since they are processed locally they are necessarily as ambiguous as the locally computed modalities that are represented by them. However, the data format the primitives provide allows for the definition of a set *semantic relations* upon them (see figure 17a). Since the primitives are a symbolic description of the local image patch, the relations and operation defined on the primitives provide the context in which information is processed.
The relations are used at a stage of processing after the condensation step (called early cognitive vision in [29]). More specifically, by the relations

- predictions between visual events become formulated (such as the change of a local image patch under motion or the likelihood of being part of the same collinear group) and by that the locally ambiguous information becomes, disambiguated (see section 5.2),
- the sets of primitives can become connected to higher visual entities such as 3D surfaces (section 5.3) and objects (section 5.4),
- low–order combinations of primitives become associated to robot actions such as grasping (section 5.5).

5.1 Relations and Operations defined on Primitives

Here we briefly describe the definition of four second order relations on primitives: Collinearity, rigid body motion, co-planarity and Co-colority (see also figure 17a).
Collinearity: In [41] a measure for the likelihood of two 2D primitives being part of the same collinear group $Coll(\pi_i, \pi_j)$ is defined (see figure 17a,i). This allows for the definition of a stereo constraint (see, e.g., [4, 44] that makes use of local image information as well as the semi–global context (see [41]). The collinearity constraint can naturally be extended to 3D primitives ($Coll(\Pi_i, \Pi_j)$).

[12] Note that the accuracy of the depth estimates decreases for horizontal structure. This is due to the ambiguity in reconstructing lines parallel to the epipolar line.



(a) left image (b) right image

(c) with sub–pixel localisation (d) no sub–pixel localisation

(e) with sub–pixel localisation (f) no sub–pixel localisation

Figure 16: Reconstruction of 3D–primitives in a real scenario. The two stereo images are shown in (a) and (b) (c) (resp. (d)): 2D–primitives extracted with (resp. without) sub–pixel localisation; and (e) (resp. (f)): spatial primitives reconstructed with (resp. without) sub–pixel localisation.

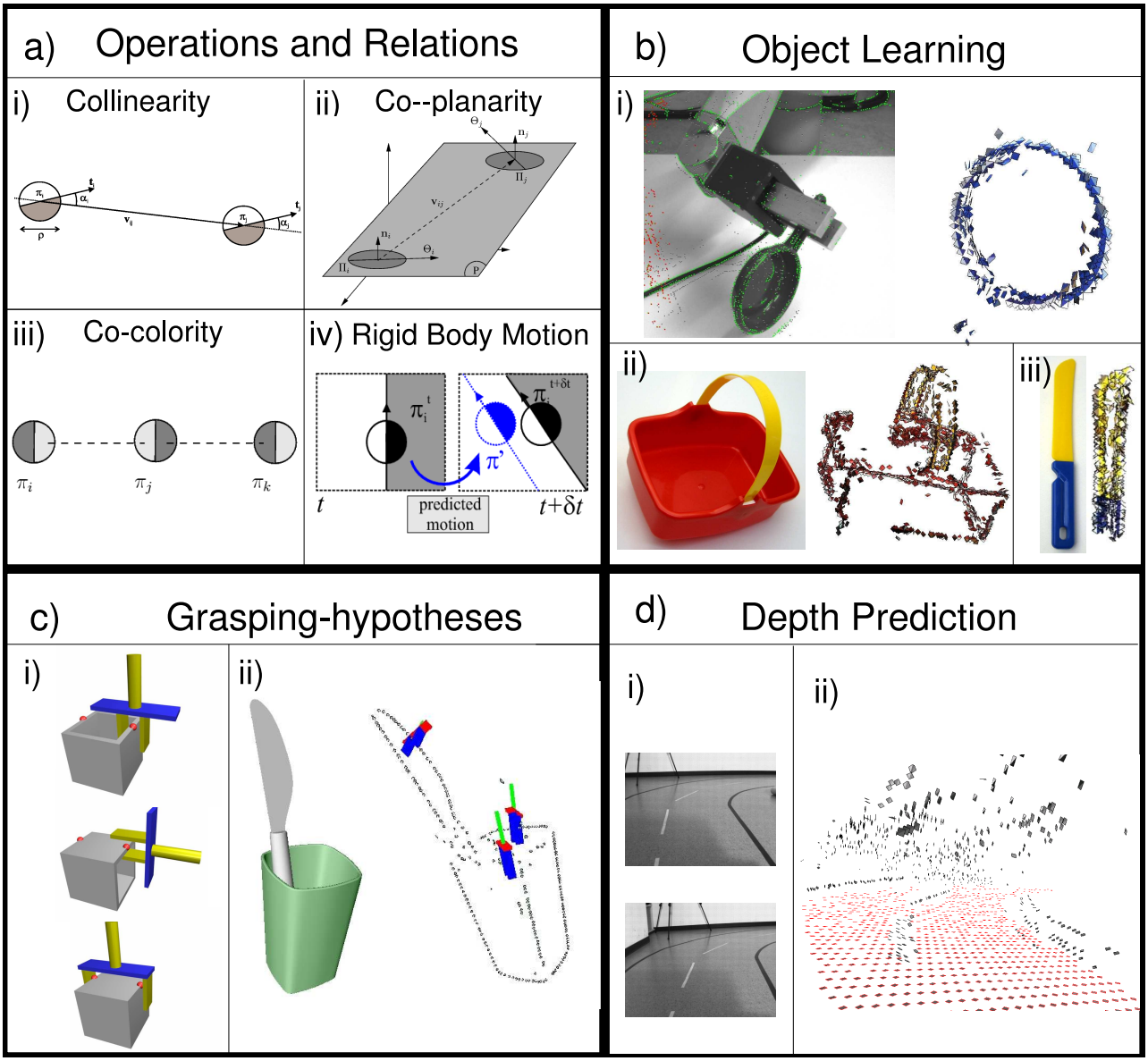


Figure 17: a) Relations defined on the multi–modal primitives. b) Grasping options generated by second order relations of primitives. c) Extraction of object representations. d) Depth predictions based on co–planarity relations.

Rigid body motion: The change of the parameters position and orientation under a rigid body motion ($RBM(\Pi)$) can be computed analytically (see, e.g., [9]) while the parameters phase and colour can be approximated to be constant under a motion (see figure 17a,iv).
Co–planarity: The relations co–planarity $Cop(\Pi_i, \Pi_j)$ between two 3D primitives (see figure 17a,ii) indicates the likelihood of the primitives to be part of the same surface (see section 5.3) and by this can be related to a grasping option (see section 5.5).
Co-colority: The relation co–colority (see figure 17a,iii) expresses the similarity of the colours at the side of two edges that are pointing towards each other.

5.2 Disambiguation using Motion and perceptual Grouping

In [42] it has been shown that such a representation allowed for computing the ego–motion of the camera rig with an accuracy sufficient for tracking individual primitives over time. It was discussed in [49] that the knowledge of this motion allows to predict the transformation between representations of the same scene at different instants, thereby correct the scene representation over time. The 3D–

hypotheses that are confirmed over time by the estimated motion gain a stronger confidence whereas hypotheses that are contradicted can be discarded as outliers.

5.3 Depth prediction at homogeneous image Areas

The primitives introduced in here represent i1D structures. It is known that it becomes increasingly difficult to find correspondences between local patches the more they are lacking structure (i.e., tending toward the i0D corner of the iD triangle (see figure 3). On the other hand, it is known that the lack of structure also indicates the lack of a depth discontinuity [17, 23]. However it was statistically shown in [24] that coplanarity allows to predict depth at homogeneous image surfaces (see figure 17d).

5.4 Object Learning and Recognition

The primitives are rich and condensed descriptors of scene information. Hence they are suitable for memorising objects in an efficient way, In particular the relations $RBM(\Pi)$ can be used to (1) get a disambiguated and hence reliable representation of objects (see section 5.2) and (2) to segment an object from the background (see figure 17b. This second property is in particular relevant in the context of the European project PACO+ [1] in which the early cognitive vision system introduced here will be linked with an AI planning system that requires objects as discrete entities (see [15]). Hence, a cognitive robot vision system should be able to find out about the 'objectness' of a set of visual features as well as the shape of the object by itself. This is achieved by combining the object learning introduced described here with the grasping approach described in section 5.5. Once representations of objects are extracted that way they can be used for pose estimation and object recognition [7].

5.5 Generating Grasping Hypotheses

Also, in the European project [1] our primitive representation is used to define grasping options in a scene (see figure 17c) and [2]). Essentially, co-planar primitives (supported by the relations co–linearity and co-colority) define planes that are good candidates for an initial grasping hypothesis. In figure 17c,i) the definition of grasping hypotheses from co-planar primitives is shown. Figure 17c,i) shows generated grasps at scenario created by the grasping simulation software GraspIt used for the evaluation of our approach (for details, see [2]). Once evaluated as successful by haptic information, gives the physical control over objects required for the object learning sketched in section 5.4.

6 Summary and Discussion

At the current state of development our system treats different scales independently. Since we are dealing with edge like structures which tend to show stable properties over different scales that is appropriate. However, it would be advantageous to find the appropriate scale to reduce memory and computational requirements. A treatment of our approach in a scale–space approach where the scale itself expressed by a feature (see, e.g., [34]) is currently being considered.
Furthermore, we intend introduce symbolic descriptors for different image structures. For homogeneous image patches this has been already discussed in section 5.3. In [50] we have discussed an extenofn opf our approach to junction–like structures. We note that this requires not only a junction detection and interpretation algorithm but also the definition of appropriate relations between different junctions as well as between edges and junctions. We are also doing first steps towards the representation of texture which in particular requires a representation of different scales.

Acknowledgement: The work on the multi–modal primitives started in 1998 in Kiel, Germany. It became an important part of the European project ECOVISION (2001–2003) [8] and is now applied in the context of vision based robotics as well as driver assistant systems in the two European projects PACOplus (2006–2010) [1] and Drivsco (2006–2009) [18]. Many master and PhD students have been involved in this project and we would like to thank Markus Ackermann, Emre Baseski, Kord Ehmcke,

Michael Felsberg, Christian Gebken, Oliver Granert, Danial Grest, Marco Hahn, Thomas Jäger, Sinan Kalkan, Dirk Kraft, Florian Pilz, Martin Pörksen, Torge Rabsch, Bodo Rosenhahn, Morten Skov, Shi Yan, Daniel Wendorff and Jan Woetzel. We would like to thank in particular and for their contributions to this work.

References

[1] Pacoplus: Perception, action and cognition through learning of object-action complexes. *Integrated Project*, 2006-2010.

[2] D. Aarno, J. Sommerfeld, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger. Early Reactive Grasping with Second Order 3D Feature Relations, journal = IEEE International Conference on Robotics and Automation (ICRA), Workshop: From features to actions - Unifying perspectives in computational and robot vision, year = 2007,.

[3] H. Barlow, C. Blakemore, and J.D. Pettigrew. The neural mechanisms of binocular depth discrimination. *Journal of Physiology (London)*, 193:327–342, 1967.

[4] R.C.K. Chung and R. Nevatia. Use of monucular groupings and occlusion analysis in a hierarchical stereo system. *Computer Vision and Image Understanding*, 62(3):245–268, 1995.

[5] H.S.M. Coxeter. *Introduction to Geometry (2nd ed.)*. Wiley & Sons, 1969.

[6] J.G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by 2d visual cortical filters. *Journal of the Optical Society of America*, 2(7):1160–1169, 1985.

[7] R. Detry, N. Pugeault, N. Krüger, and J. Piater. Hierarchical integration of local 3D features for probablistic pose estimation. *INTELSIG Technical Report 2007–01–19, Department of Electrical Engineering and Computer Science University of Liege*, 2007.

[8] ECOVISION. Artificial visual systems based on early-cognitive cortical processing (EU–Project). *http://www.pspc.dibe.unige.it/ecovision/project.html*, 2003.

[9] O.D. Faugeras. *Three–Dimensional Computer Vision*. MIT Press, 1993.

[10] M. Felsberg. *Low-Level Image Processing with the Structure Multivector*. PhD thesis, Institute of Computer Science and Applied Mathematics, Christian-Albrechts-University of Kiel, 2002.

[11] M. Felsberg, S. Kalkan, and N. Krüger. Continuous characterization of image structures of different dimensionality. in preparation.

[12] M. Felsberg and N. Krüger. A probablistic definition of intrinsic dimensionality for images. *Pattern Recognition, 24th DAGM Symposium*, 2003.

[13] M. Felsberg and G. Sommer. A new extension of linear signal processing for estimating local properties and detecting features. In G. Sommer, N. Krüger, and C. Perwass, editors, *22. DAGM Symposium Mustererkennung, Kiel*, pages 195–202. Springer-Verlag, Heidelberg, 2000.

[14] M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 49(12):3136–3144, December 2001.

[15] Ch. Geib, K. Mourao, R. Petrick, N. Pugeault, M. Steedman, N. Krüger, and F. Wörgötter. Object action complexes as an interface for planning and robot control. *Workshop 'Toward Cognitive Humanoid Robots' at IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, 2006.

[16] G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision*. Kluwer Academic Publishers, Dordrecht, 1995.

[17] W.E.L. Grimson. Surface consistency constraints in vision. *CVGIP*, 24(1):28–51, 1983.

[18] http://www.pspc.dibe.unige.it/ drivsco/, editor. *DRIVSCO: Learning to Emulate Perception-Action Cycles in a Driving School Scenario (FP6-IST-FET, contract 016276-2)*. 2006-2009.

[19] D.H. Hubel and T.N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Phyiology*, 160:106–154, 1962.

[20] D.H. Hubel and T.N. Wiesel. Anatomical demonstration of columns in the monkey striate cortex. *Nature*, 221:747–750, 1969.

[21] B. Jähne. *Digital Image Processing – Concepts, Algorithms, and Scientific Applications*. Springer, 1997.

[22] S. Kalkan, D. Calow, F. Wörgötter, M. Lappe, and N. Krüger. Local image structures and optic flow estimation. *Network: Computation in Neural Systems*, 16(4):341–356, 2005.

[23] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of local 3d structure in 2d images. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1114–1121, 2006.

[24] S. Kalkan, F. Wörgötter, and N. Krüger. Statistical analysis of second-order relations of 3d structures. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[25] P. König and N. Krüger. Perspectives: Symbols as self-emergent entities in an optimization process of feature extraction and predictions. *Biological Cybernetics*, 94(4):325–334, 2006.

[26] P. Kovesi. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3):1–26, 1999.

[27] N. Krüger and M. Felsberg. A continuous formulation of intrinsic dimension. *Proceedings of the British Machine Vision Conference*, pages 261–270, 2003.

[28] N. Krüger and M. Felsberg. An explicit and compact coding of geometric and structural information applied to stereo matching. *Pattern Recognition Letters*, 25(8):849–863, 2004.

[29] N. Krüger, M. Van Hulle, and F. Wörgötter. Ecovision: Challenges in early-cognitive vision. *International Journal of Computer Vision*, submitted.

[30] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 1(5):417–428, 2004.

[31] N. Krüger and F. Wörgötter. Multi modal estimation of collinearity and parallelism in natural image sequences. *Network: Computation in Neural Systems*, 13:553–576, 2002.

[32] N. Krüger and F. Wörgötter. Statistical and deterministic regularities: Utilisation of motion and grouping in biological and artificial visual systems. *Advances in Imaging and Electron Physics*, 131:82–147, 2004.

[33] M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R.P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamik link architecture. *IEEE Transactions on Computers*, 42(3):300–311, 1993.

[34] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998.

[35] D. Marr. *Vision: A computational investigation into the human representation and processing of visual information*. Freeman, 1977.

[36] R.M. Mersereau. The processing of hexagonally sampled two–dimensional signals. *Proc. IEEE*, 67(6):930–949, 1979.

[37] L. Middleton and J. Sivaswamy. *Hexagonal Image Processing : A Practical Approach*. Springer Verlag, 2005.

[38] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:565–593, 1986.

[39] M.W. Oram and D.I. Perrett. Modeling visual recognition from neurobiological constraints. *Neural Networks*, 7:945–972, 1994.

[40] N. Pugeault. *Working Title: Early Cognitive Vision*. 2006.

[41] N. Pugeault, F. Wörgötter, , and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proceedings of the 5th IEEE Computer Society Workshop on Perceptual Organization in Computer Vision, New York City June 22, 2006 (in conjunction with IEEE CVPR 2006)*, 2006.

[42] N. Pugeault, F. Wörgötter, , and N. Krüger. Rigid body motion in an early cognitive vision framework. In *Proceedings of the IEEE Systems, Man and Cybernetics Society Conference on Advances in Cybernetic Systems*, 2006.

[43] S.P. Sabatini, Karl, Javier, and Nico. Working title: Analysis of harmonic filter design. to be submitted.

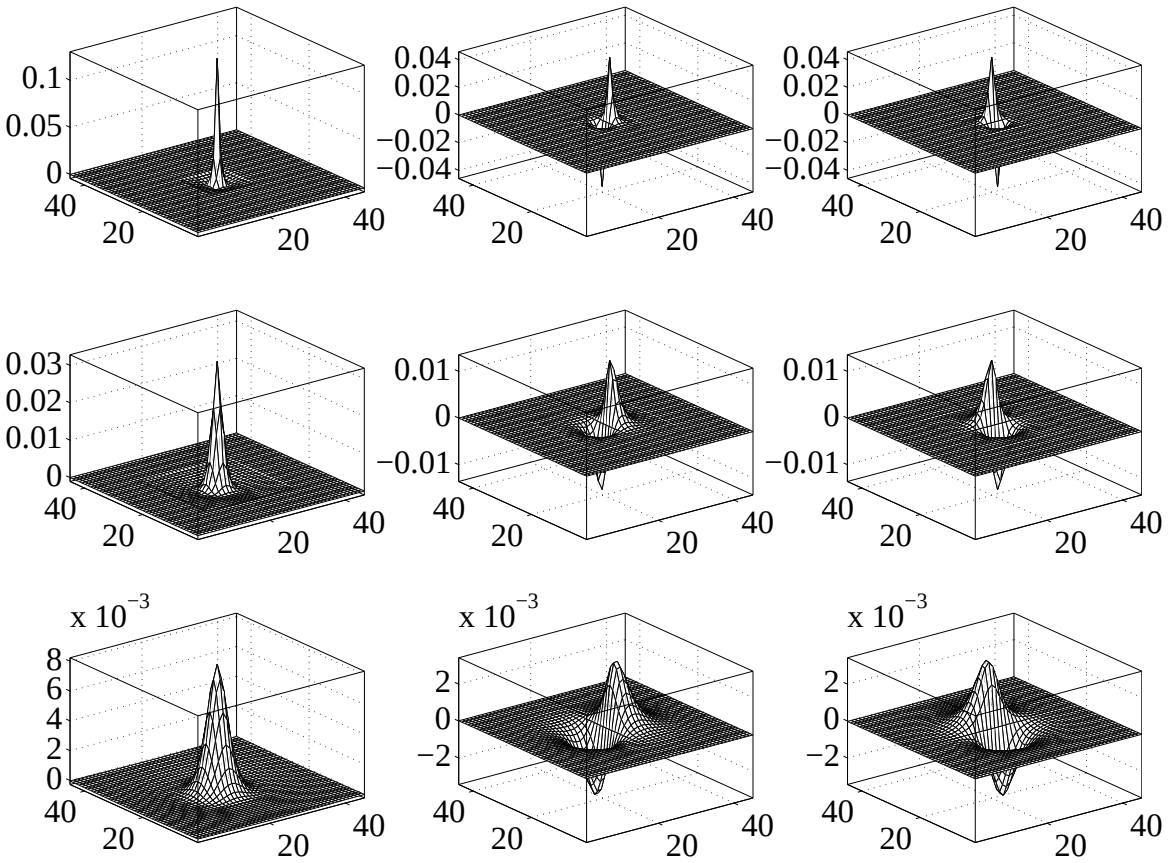


Figure 18: Impulse responses of the DOP filter and its Riesz transforms. From left to right: DOP filter, first Riesz transform, second Riesz transform. From top to bottom: scales (1,2), (2,4), (4,8).

[44] S. Sarkar and K.L. Boyer. *Computing Perceptual Organization in Computer Vision*. World Scientific, 1994.

[45] B. Schiele and J.L. Crowley. Probabilistic object recognition using multidimensional receptive field histograms. *Advances in Neural Information Processing Systems*, 8:865–871, 1996.

[46] I.A. Shevelev, N.A. Lazareva, A.S. Tikhomirov, and G.A. Sharev. Sensitivity to cross–like figures in the cat striate neurons. *Neuroscience*, 61:965–973, 1995.

[47] R.C. Staunton and N. Storey. A comparison between square and hexagonal sampling methods for pipeline image processing. *Proc. SPIE*, 1194:142–151,, 1989.

[48] Lawrence B. Wolff. Accurate measurements of orientation from stereo using line correspondence. 1989.

[49] F. Wörgötter, N. Krüger, N. Pugeault, D. Calow, M. Lappe, K. Pauwels, M. Van Hulle, S. Tan, and A. Johnston. Early cognitive vision: Using gestalt-laws for task-dependent, active image-processing. *Natural Computing*, 3(3):293–321, 2004.

[50] Shi Yan, Sinan Kalkan, Nicolas Pugeault, and Norbert Krüger. Corner stuff. to be submitted.

[51] C. Zetzsche and E. Barth. Fundamental limits of linear filters in the visual processing of two dimensional signals. *Vision Research*, 30, 1990.

A Split of Identity

Quadrature filters based on the monogenic signal [14] are rotation invariant, i.e., they commute with the rotation operator. Hence, for an appropriate choice of polar coordinates, two coordinates do

not change under rotations (amplitude and phase), whereas the third coordinate directly reflects the rotation angle. This kind of quadrature filter, which is called *spherical quadrature filter* [10], is formed by triplet of filters: a radial bandpass filter and its two Riesz transforms [21]. As in [10] we construct the bandpass filter from *difference of Poisson* (DOP) filters, in order to get analytic formulations of all filter components in the spatial domain and in the frequency domain. The DOP filter is an even filter (w.r.t. point reflections in the origin) and its impulse response (convolution kernel) and frequency response (Fourier transform of the kernel) are respectively given by:

$$h_{\mathrm{e}}(\boldsymbol{x}) = \frac{s_1}{2\pi(|\boldsymbol{x}|^2+s_1^2)^{\frac{3}{2}}} - \frac{s_2}{2\pi(|\boldsymbol{x}|^2+s_2^2)^{\frac{3}{2}}} \tag{13}$$

$$H_{\mathrm{e}}(\boldsymbol{u}) = \exp(-2\pi|\boldsymbol{u}|s_1) - \exp(-2\pi|\boldsymbol{u}|s_2) \ . \tag{14}$$

For convenience, we combine the two Riesz transforms of the DOP filter in a complex, odd filter, yielding the impulse response and the frequency response:

$$h_{\mathrm{o}}(\boldsymbol{x}) = \frac{\boldsymbol{x}_1+i\boldsymbol{x}_2}{2\pi(|\boldsymbol{x}|^2+s_1^2)^{\frac{3}{2}}} - \frac{\boldsymbol{x}_1+i\boldsymbol{x}_2}{2\pi(|\boldsymbol{x}|^2+s_2^2)^{\frac{3}{2}}} \tag{15}$$

$$H_{\mathrm{o}}(\boldsymbol{u}) = \frac{u_2-iu_1}{|\boldsymbol{u}|}(\exp(-2\pi|\boldsymbol{u}|s_1) - \exp(-2\pi|\boldsymbol{u}|s_2)) \ , \tag{16}$$

respectively. The impulse responses of the filters for $(s_1, s_2) = (1,2), (2,4), (4,8)$ are shown in figure 18. The split of identity (i.e., the separation of the signal into local amplitude, orientation and phase) is obtained by switching to appropriate polar coordinates. In particular, we transform the filter responses according to

$$m(\boldsymbol{x}) = \sqrt{I_{\mathrm{e}}(\boldsymbol{x})^2 + |I_{\mathrm{o}}(\boldsymbol{x})|^2} \tag{17}$$

$$\theta(\boldsymbol{x}) = \arg I_{\mathrm{o}}(\boldsymbol{x}) \pmod{\pi} \tag{18}$$

$$\varphi(\boldsymbol{x}) = sign(\Im\{I_{\mathrm{o}}(\boldsymbol{x})\}) \arg(I_{\mathrm{e}}(\boldsymbol{x}) + i|I_{\mathrm{o}}(\boldsymbol{x})|) \ , \tag{19}$$

which gives the desired amplitude, orientation, and phase information.
Figure 19 shows a radial cut through the DOP bandpass filters for a certain range of scales and their superposition, demonstrating a homogeneous covering of the frequency domain. For infinitely many bandpass filters, the superposition is one everywhere, except at the origin. In our system, we apply filters on three frequency levels (see figure 18). The applied bandpasses are indicated by the darker colour in figure 19.

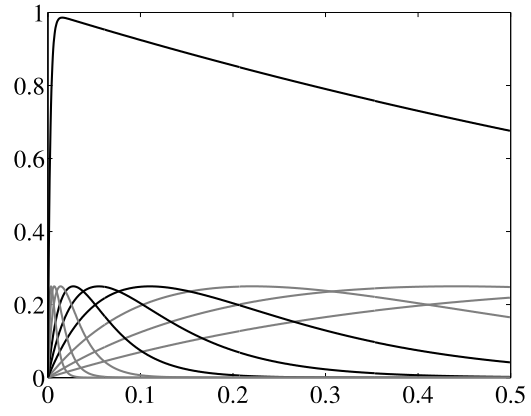


Figure 19: DOP bandpass filters and their superposition approaching the identity (x–axis representing the frequency). The superposition and the filters applied in this paper are indicated by the darker lines.

Extraction of multi–modal Object representations in a Robot Vision System

International Workshop on Robot Vision, in conjunction with VISAPP'07

Nicolas Pugeault[1], Emre Baseski[2], Dirk Kraft[2], Florentin Wörgötter[3], and Norbert Krüger[2]

[1] University of Edinburgh,
United Kingdom.
npugeaul@inf.ed.ac.uk
[2] Syddansk Universitet,
Denmark.
emre,kraft,norbert@mip.sdu.dk
[3] Göttingen University,
Germany.
worgott@jupiter.chaos.gwdg.de

Abstract. We introduce one module in a cognitive system that learns the shape of objects by active exploration. More specifically, we propose a feature tracking scheme that makes use of the knowledge of a robotic arm motion to: 1) segment the object currently grasped by the robotic arm from the rest of the visible scene, and 2) learn a representation of the 3D shape without any prior knowledge of the object. The 3D representation is generated by stereo–reconstruction of local multi–modal edge features. The segmentation between features belonging to the object those describing the rest of the scene is achieved using Bayesian inference. We then show the shape model extracted by this system from various objects.

1 Introduction

A cognitive robot system should to be able to extract representations about its environment by exploration to enrich its internal representations and by this its cognitive abilities (see, e.g., [4]). The knowledge about the existence of objects and their shapes is of particular importance in this context. Having a model of an object that includes 3D information allows for the recognition and finding of poses of objects (see, e.g., [9]) as well as grasp planning (e.g. [1], [10]). However, extracting such representations of objects has shown to be very difficult. Hence many systems are based on CAD models or other manually achieved information. In this paper, we introduce a module that extracts multi–modal representations of objects by making use of the interaction of a grasping system with an early cognitive vision system (see Fig. 1 and [7]). After gaining physical control over an object (for example by making use of the object-knowledge independent grasping strategy in [2]) it is possible to formulate predictions about the change of rich feature description under the object motion induced by the robot.

If the motions of the objects within the scene are known, then the relation between features in two subsequent frames becomes deterministic (excluding the usual problems of occlusion, sampling, etc.). This means that a structure (e.g. in our case a contour) that is present in one frame is guaranteed to be in the previous and next frames (provided it does not become occluded or goes out of the field of view of the camera), subject a transformation that is fully determined by the motion: generally a change of position and orientation. If we assume that the motions are reasonably small compared to the frame–rate, then a contour will not appear or disappear unpredictably, but will have a life–span in the representation, between the moment it entered the field of view and the moment it leaves it (partial or complete occlusion may occur during some of the time–steps).
These prediction are relevant in different contexts

- **Establishment of objectness:** The objectness of a set of features is characterised by the fact that they all move according to the robot motion. This property is discussed in the context of a grounded AI planning system in [5].
- **Segmentation:** The system segments the object by its predicted motion from the other parts of the scene.
- **Disambiguation:** Ambiguous features can be characterised (and eliminated) by not moving according to the predictions.
- **Learning of object model:** A full 3D model of the object can be extracted by merging different views created by the motion of the end effector.

In this work, we represent objects as sets of multi–modal visual descriptors called 'primitives' covering visual information in terms of geometric 3D information (position and orientation) as well as appearance information (colour and phase). This representation is briefly described in section 2. The predictions based on rigid motion are described in section 3. The predictions are then used to track primitives over frames and to accumulate likelihoods for the existence of features (section 4). This is formulated in a Bayesian framework in section 4.3. In section 5, we finally show results of object acquisition for different objects and scenes.

2 Introducing visual primitives

The primitives we will be using in this work are local, multi–modal edge descriptors that were introduced in [8] (see figure 1). In contrast to the above mentioned features these primitives focus on giving a semantically and geometrically meaningful description of the local image patch. The importance of such a semantic grounding of features for a general purpose vision front–end, and the relevance of edge–like structures for this purposes were discussed in [3].
The primitives are extracted sparsely at locations in the image that are the most likely to contain edges. The sparseness is assured using a classical winner take all operation, insuring that the generative patches of the primitives do not overlap. Each of the primitive encodes the image information contained by a local image patch. Multi–modal information is gathered from this image patch, including the position $\boldsymbol{x}$ of the centre of the patch, the orientation θ of the edge, the phase ω of the signal at this point, the colour $\boldsymbol{c}$ sampled over the image patch on both sides of the edge, the local optical flow $\boldsymbol{f}$ and the size of the patch ρ. Consequently a local image patch is described by the following multi–modal vector:

$$\boldsymbol{\pi} = (\boldsymbol{x}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T, \tag{1}$$

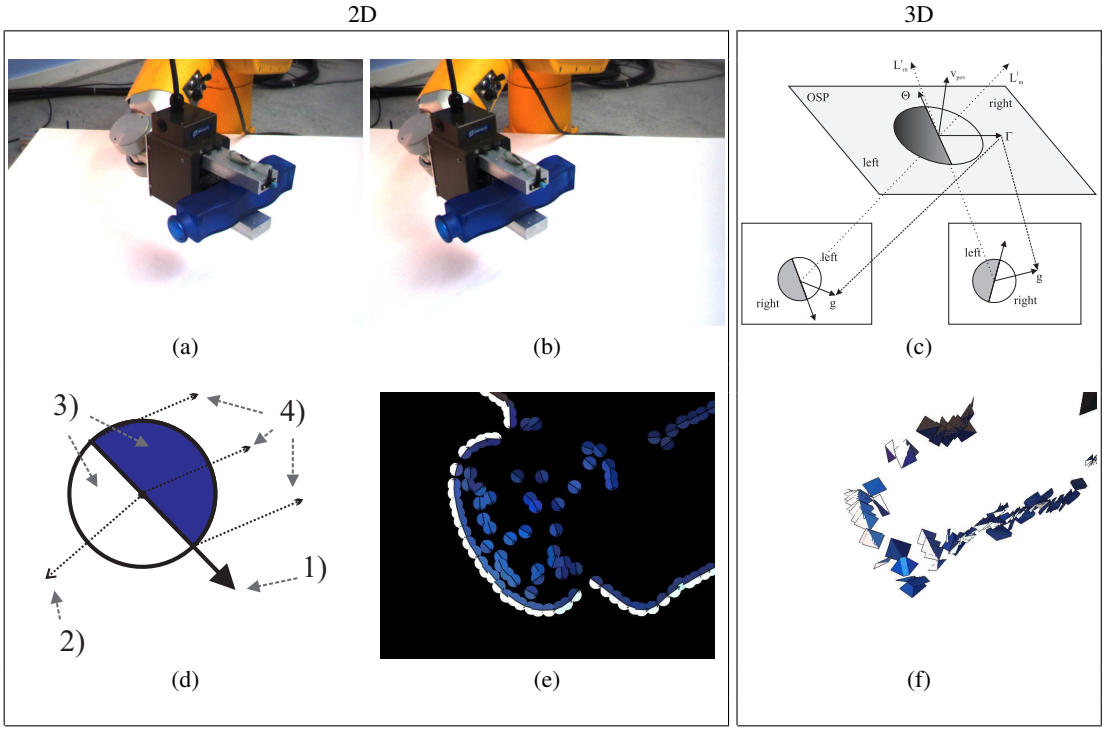


Fig. 1. Overview of the system. (a)-(b) images of the scene as viewed by the left and right camera at the first frame. (d) symbolic representation of a primitive: wherein 1) shows the orientation, 2) the phase, 3) the colour and 4) the optic flow of the primitive. (e) 2D–primitives of a detail of the object. (c) reconstruction of a 3D–primitive from a stereo–pair of 2D–primitives. (f) 3D–primitives reconstructed from the scene.

that we will name *2D primitive* in the following. The primitive extraction process is illustrated in Fig. 1.

Note that these primitives are of lower dimensionality than, e.g., SIFT (10 vs. 128) and therefore suffer of a lesser distinctiveness. Nonetheless, as shown in [11], they are distinctive enough for a reliable stereo matching if the epipolar geometry of the cameras is known. Furthermore, their semantic in terms of geometric and appearance based information allow for a good description of the scene content.

In a stereo scenario *3D primitives* can be computed from correspondences of 2D primitives (see Fig.1)

$$\boldsymbol{\Pi} = (\boldsymbol{X}, \boldsymbol{\Theta}, \Omega, \boldsymbol{C})^T, \tag{2}$$

where $\boldsymbol{X}$ is the position in space, $\boldsymbol{\Theta}$ is the 3D orientation, Ω is the phase of the contour and $\boldsymbol{C}$ is the colour on both sides of the contour. We have a projection relation

$$\mathcal{P} : \boldsymbol{\Pi} \rightarrow \boldsymbol{\pi} \tag{3}$$

linking 3D–primitives and 2D–primitives.

We call scene representation $\boldsymbol{\mathcal{S}}$ the set of all 3D–primitives reconstructed from a stereo–pair of images.

3 Making predictions from the Robot Motion

If we consider a 3D–primitive $\boldsymbol{\Pi}_i^t \in \boldsymbol{\mathcal{S}}_t$ part of the scene representation at an instant t, and assuming that we know the motion of the objects between two instants t and $t + \Delta t$, we can predict the position of the primitive in the new coordinate system of the camera at $t + \Delta t$.
Concretely, we predict the scene representation $\boldsymbol{\mathcal{S}}_{t+\Delta t}$ by moving the anterior scene representation ($\boldsymbol{\mathcal{S}}_t$) according to the estimated motion between instants t and $t + \Delta t$. The mapping $\mathcal{M}_{t \to t+\Delta t}$ associating the any entity in space in the coordinate system of the stereo set–up at time t to the same entity in the new coordinate at time $t + \Delta t$ is explicitly defined for 3D–primitives:

$$\hat{\boldsymbol{\Pi}}_i^{t+\Delta t} = \mathcal{M}_{t \to t+\Delta t}(\boldsymbol{\Pi}_i^t) \tag{4}$$

Assuming a scene representation $\boldsymbol{\mathcal{S}}_t$ is correct, and that the motion between two instants t and $t + \Delta t$ is known, then the moved representation $\hat{\boldsymbol{\mathcal{S}}}_{t+\Delta t}$ according to the motion $\mathcal{M}_{t \to t+\Delta t}$ is a *predictor* for the scene representation $\boldsymbol{\mathcal{S}}_{t+\Delta t}$ that can be extracted by stereopsis at time $t + \Delta t$.

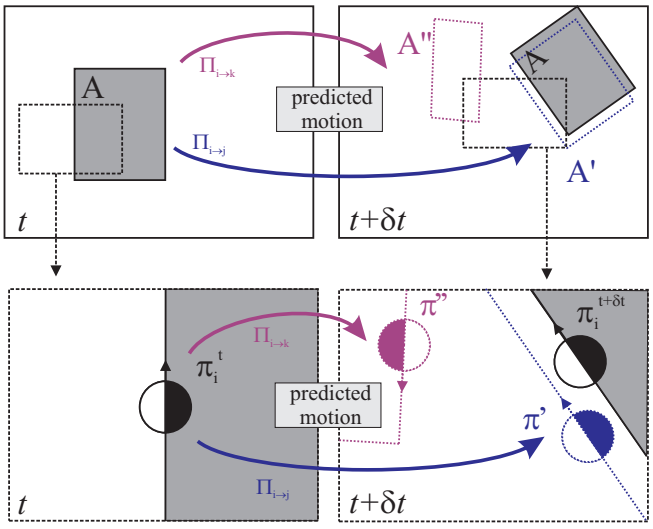


Fig. 2. Example of the accumulation of a primitive (see text).

Note that the predicted representation stems from the primitives extracted from the cameras at time t whereas the real scene representation is issued from primitives extracted at time $t + \Delta t$.
By extension, this relation also applies to the image representations reprojected onto each of the stereo image planes $\boldsymbol{\mathcal{I}}^F$, $F \in \{\text{left,right}\}$, defined by a projection $\mathcal{P}^F$:

$$\hat{\boldsymbol{\pi}}_i^{F,t+\Delta t} = \mathcal{P}^F\left(\mathcal{M}_{t \to t+\Delta t}(\boldsymbol{\Pi}_i^t)\right) \tag{5}$$

This prediction/verification process is illustrated in Fig. 2. The left column shows the image at time t whereas the right column shows the image at time $t + \Delta t$. The top row shows the complete image of the object and the bottom row shows

details of the object specified by the black rectangle. If we consider the object **A** with (solid rectangle in the top–left and top–right images) that between time t and $t + \Delta t$ according to a motion $M_{t \rightarrow t+\Delta t}$ Two hypotheses on the 3D shape of the object lead to two distinct predictions at time $t + \Delta t$: **A'** (correct and close to the actual pose of the object, blue rectangle in the top–right image) and **A"** (erroneous, red rectangle). In the bottom row, we study the case of a specific 2D–primitive $\boldsymbol{\pi}_i^t$ lying on the contour of **A** at the instant t (bottom–left image). If one consider that, at time t, there was two ambiguous stereo correspondences $\boldsymbol{\pi}_j^t$ and $\boldsymbol{\pi}_k^t$ then we have two mutually exclusive 3D reconstructions $\boldsymbol{\Pi}_{i \rightarrow j}^t$ and $\boldsymbol{\Pi}_{i \rightarrow k}^t$, each predicting a different pose at time $t + \Delta t$: 1) the correct hypothesis $\boldsymbol{\Pi}_{i \rightarrow j}^t$ predicts a 2D–primitive $\boldsymbol{\pi}'$ that matches with $\boldsymbol{\pi}_i^{t+\Delta t}$ (blue in the bottom–right image), one of the a 2D–primitive newly extracted at $t + \Delta t$ from the contour of **A**, comforting the original hypothesis; 2) when moving the incorrect hypothesis $\boldsymbol{\Pi}_{i \rightarrow k}^t$ we predict a 2D–primitive $\boldsymbol{\pi}''$ (red in the bottom–right image), that do not match any primitive extracted from the image, thereby revealing the erroneousness of the hypothesis.

Differences in viewpoint and pixel sampling lead to large variation in the primitives extracted and the resulting stereopsis. In other words, this means that the same contours of the scene will be described in the image representation, but by slightly shifted primitives, sampled at different points, along these contours. Therefore we need to devise a tracking algorithm able to recognise similar structures between heterogeneous representations.[4]

If a precise robot like the Staubli RX60 is used to move the objects the motion of the robot can be used to predict the primitive positions. Hereby it needs to be mentioned that the primitive position and orientation are usually represented in the camera coordinate system (placed in the left camera) while the robot movements are relative to the robot coordinate system (for the RX60 this is located at its first joint). To compute the mapping between the two coordinate systems we use a calibration procedure in which the robot end effector is moved to the eight positions of a virtual cube. At each location the position of the end effector in both coordinate systems are noted. The transformation between the two systems can then be computed by solving the overdetermined linear equation system represented by the eight positions. We use the RBM estimation algorithm described in [12] to do this.

4 Tracking 3D-primitives over time

In this section we will address the problem of integrating two heterogeneous scene representations, one extracted and one predicted that both describe the same scene at the same instant from the same point of view. The problem is three–fold: 1) comparing the two representations, 2) including the extracted primitives that were not predicted, and 3) re–evaluating the confidence in each of the primitives according to their predictability.

[4] We note here that the transformation described in this section does not describe the change of edges for a specific class of occlusions that occurs when round surfaces become rotated. In these cases the reconstructed edges do not move according to an RBM.

4.1 2D comparison

We propose to compare the two representations in the 2D image plane domain. This can be done by reprojecting all the 3D–primitives in the predicted representation $\hat{\boldsymbol{\mathcal{S}}}_{t+\Delta t}$ onto both image planes, creating two predicted image representations

$$\hat{\boldsymbol{\mathcal{I}}}^{F}_{t+\Delta t} = \mathcal{P}^{F}\left(\hat{\boldsymbol{\mathcal{S}}}_{t+\Delta t}\right), F \in \{\text{left,right}\} \tag{6}$$

Then both predicted image representations $\hat{\boldsymbol{\mathcal{I}}}^{F}_{t+\Delta t}$ can be compared with the extracted primitives $\boldsymbol{\mathcal{I}}^{F}_{t+\Delta t}$. For each predicted primitive $\hat{\boldsymbol{\pi}}_i$, a small neighbourhood (the size of the primitive itself) is searched for an extracted primitive $\boldsymbol{\pi}_j$ whose position and orientation are very similar (with a distance less than a threshold t_θ). Effectively a given prediction $\hat{\boldsymbol{\Pi}}_i$ is labelled as matched $\mu(\hat{\boldsymbol{\Pi}}_i)$ iff. for each image plane F defined by the projection $\mathcal{P}^F$ and having an associated image representation $\boldsymbol{\mathcal{I}}^F_t$, we have the projection $\boldsymbol{\pi}^F_i = \mathcal{P}^x(\boldsymbol{\Pi}_i)$ satisfy the following relation:

$$\exists \boldsymbol{\pi}_j \in \boldsymbol{\mathcal{I}}^F_t, \begin{cases} d_{\text{2D}}(\hat{\boldsymbol{\pi}}^F_i, \boldsymbol{\pi}_j) < r_{2D}, \\ d_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\pi}}^F_i, \boldsymbol{\pi}_j) < t_{\boldsymbol{\Theta}} \end{cases} \tag{7}$$

with r_{2D} being the radius of correspondence search in pixels, $t_{\boldsymbol{\Theta}}$ being the maximal orientation error allowed for matching, d_{2D} stands for the two–dimensional Euclidian distance, and $d_{\boldsymbol{\Theta}}$ is the orientation distance. This is also illustrated in Fig. 2.

This 2D–matching approach has the following advantages: First, as we are comparing the primitives in the image plane, we are not affected by the inaccuracies and failures due to the 3D–reconstruction (see also [6]). Second, using the extracted 2D–primitives directly allows for 2D–primitives that could not be reconstructed at this time–step due to errors in stereo matching, etc.

4.2 Integration of different scene representations

Given two scene representations, one extracted $\boldsymbol{\mathcal{S}}_t$ and one predicted $\hat{\boldsymbol{\mathcal{A}}}_t$ we want to merge them into an accumulated representation $\boldsymbol{\mathcal{A}}_t$.

The application of the tracking procedure presented in section 4.1 provides a separation of the 3D–primitives in $\boldsymbol{\mathcal{S}}_t$ into three groups: confirmed, unconfirmed and not predicted.

The integration process consist into adding to the accumulated representation $\boldsymbol{\mathcal{A}}_{t-1}$, all 3D–primitives issued from the scene representation $\boldsymbol{\mathcal{S}}_t$ that are not matched by any 3D–primitive in $\boldsymbol{\mathcal{A}}_{t-1}$ (*i. e.* the non–predicted ones).

$$\boldsymbol{\mathcal{A}}_t = \boldsymbol{\mathcal{A}}_{t-1} \cup \boldsymbol{\mathcal{S}}_t \tag{8}$$

This allows to be sure that the accumulated representation always strictly include the newly extracted representation ($\boldsymbol{\mathcal{S}}_t \subseteq \boldsymbol{\mathcal{A}}_t$), and enables to include new information in the representation.

4.3 Confidence re–evaluation from tracking

The second mechanism allows to re–evaluate the confidence in the 3D–hypotheses depending on their resilience. This is justified by the continuity assumption, which

states that 1) any given object or contour of the scene should not appear and disappear in and out of the field of view (FoV) but move gracefully in and out according to the estimated ego–motion, and 2) that the position and orientation of such a contour at any point in time is fully defined by the knowledge of its position at a previous point in time and of the motion of this object between these two instants.

As we exclude from this work the case of independent moving object, and as the ego–motion is known, all conditions are satisfied and we can trace the position of a contour extracted at any instant t at any later stage $t + \Delta t$, as well as predict the instant when it will disappear from the FoV.

We will write the fact that a primitive $\boldsymbol{\Pi}_i$ that predicts a primitive $\hat{\boldsymbol{\Pi}}_i^t$ at time t is matched (as described above) as $\mu_t(\hat{\boldsymbol{\Pi}}_i)$ We define the tracking history of a primitive $\boldsymbol{\Pi}_i$ from its apparition at time 0 until time t as:

$$\boldsymbol{\mu}(\boldsymbol{\Pi}_i) = \left(\mu_t(\hat{\boldsymbol{\Pi}}_i), \mu_{t-1}(\hat{\boldsymbol{\Pi}}_i), \cdots, \mu_0(\hat{\boldsymbol{\Pi}}_i)\right)^T \tag{9}$$

thus, applying Bayes formula:

$$p\left(\boldsymbol{\Pi}_i|\boldsymbol{\mu}(\hat{\boldsymbol{\Pi}}_i)\right) = \frac{p\left(\boldsymbol{\mu}(\hat{\boldsymbol{\Pi}}_i)|\boldsymbol{\Pi}\right) p\left(\boldsymbol{\Pi}\right)}{p\left(\boldsymbol{\mu}(\hat{\boldsymbol{\Pi}}_i)|\boldsymbol{\Pi}\right) p\left(\boldsymbol{\Pi}\right) + p\left(\bar{\boldsymbol{\mu}}(\hat{\boldsymbol{\Pi}}_i)|\bar{\boldsymbol{\Pi}}\right) p\left(\bar{\boldsymbol{\Pi}}\right)} \tag{10}$$

where $\boldsymbol{\Pi}$ and $\bar{\boldsymbol{\Pi}}$ are correct and erroneous primitives, respectively.

Furthermore, if we assume independence between the matches we have, and assuming that $\boldsymbol{\Pi}$ exists since n iterations and has been matched successfully m times, we have:

$$\begin{aligned} p\left(\boldsymbol{\mu}(\hat{\boldsymbol{\Pi}}_i)|\boldsymbol{\Pi}\right) &= \textstyle\prod_t p\left(\mu_t(\hat{\boldsymbol{\Pi}}_i)|\boldsymbol{\Pi}\right) \\ &= p\left(\mu_t(\hat{\boldsymbol{\Pi}}_i) = 1|\boldsymbol{\Pi}\right)^m p\left(\mu_t(\hat{\boldsymbol{\Pi}}_i) = 0|\boldsymbol{\Pi}\right)^{n-m} \end{aligned} \tag{11}$$

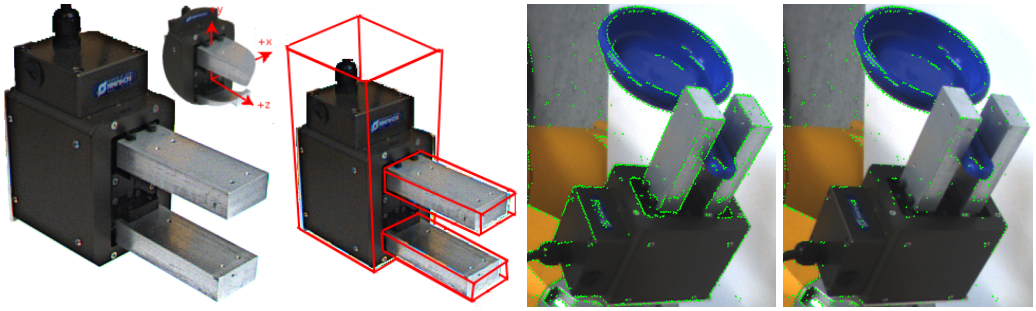
In this case the probabilities for μ_t are equiprobable for all t, and therefore we define the quantities $\alpha = p\left(\boldsymbol{\Pi}\right)$, $\beta = p\left(\mu_t(\hat{\boldsymbol{\Pi}}) = 1|\boldsymbol{\Pi}\right)$ and $\gamma = p\left(\mu_t(\hat{\boldsymbol{\Pi}}) = 1|\bar{\boldsymbol{\Pi}}\right)$ then we can rewrite (10) as follows:

$$p\left(\boldsymbol{\Pi}_i|\bar{\boldsymbol{\mu}}(\hat{\boldsymbol{\Pi}}_i)\right) = \frac{\beta^m(1-\beta)^{n-m}\alpha}{\beta^m(1-\beta)^{n-m}\alpha + \gamma^m(1-\gamma)^{n-m}(1-\alpha)} \tag{12}$$

We measured these prior and conditional probabilities using a video sequence with known motion and depth ground truth obtained via range scanner. We found values of $\alpha = 0.46$, $\beta = 0.83$ and $\gamma = 0.41$. This means that, in these examples, the prior likelihood for a stereo hypothesis to be correct is 46%, the likelihood for a correct hypothesis to be confirmed is 83% whereas for an erroneous hypothesis it is of 41%. These probabilities show that Bayesian inference can be used to identify correct correspondences from erroneous ones. To stabilise the process, we will only consider the n first frames after the appearance of a new 3D–primitive. After n frames, the confidence is fixed for good. If the confidence is deemed too low at this stage, the primitive is forgotten. During our experiments $n = 5$ proved to be a suitable value.

4.4 Eliminating the grasper

The end-effector of the robot follows the same motion as the object. Therefore, this end-effector becomes extracted as well. Since we know the geometry of this end-effector (Figure 3 (a)), we can however easily subtract it by eliminating the 3D primitives that are inside the bounding boxes that bounds the body of the gripper and its fingers (Figure 3 (b)). For this operation, three bounding boxes are calculated in grasper coordinate system (GCS) by using the dimensions of grasper. Since the 3D primitives are in robot coordinate system (RCS), the transformation from RCS to GCS is applied to each primitive and if the resultant coordinate is inside any of the bounding boxes, the primitive is eliminated. In Figure 3 (c) 2D projection of 3D primitives extracted from a stereo pair is presented. After gripper elimination, 2D projection of remaining primitives are shown in Figure 3 (d).



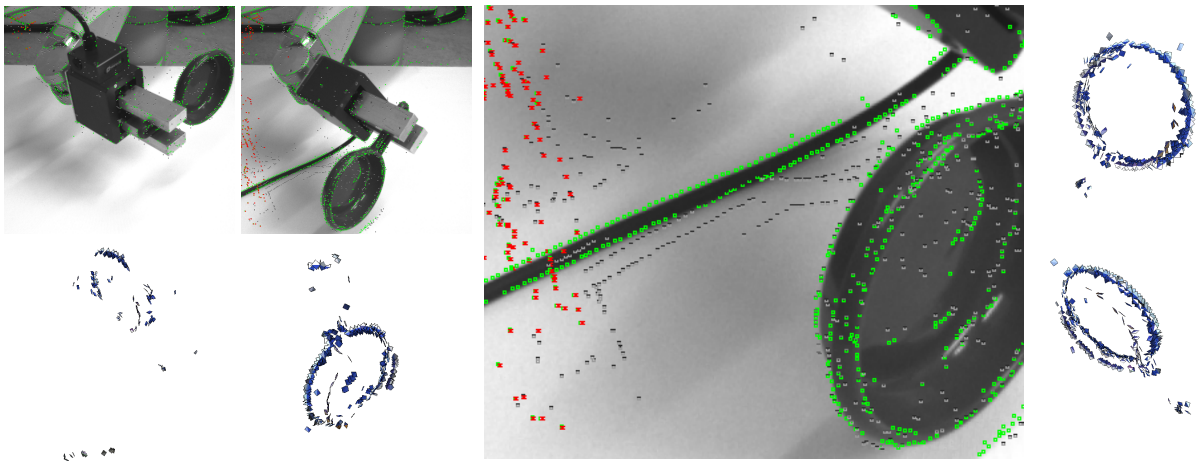
(a) (b) (c) (d)

Fig. 3. Gripper elimination (a) grasper and grasper coordinate system (b) bounding boxes of grasper body and its fingers (c) primitives before grasper elimination (d) primitives after grasper elimination

5 Results and Conclusion

We applied the accumulation scheme to a variety of scenes where the robot arm manipulated several objects. The motion was a rotation of 5 degrees per frame. The accumulation process on one such object is illustrated in Fig. 4. The top row show the predictions at each frame. The bottom row, shows the 3D–primitives that were accumulated (frames 1, 12, 22, and 32). The object representation becomes fuller over time, whereas the primitives reconstructed from other parts of the scene are discarded. Figure 5 shows the accumulated representation for various objects. The hole in the model corresponds to the part of the object occluded by the gripper. Accumulating the representation over several distinct grasps of the objects would yield a complete representation.

Conclusion: In this work we presented a novel scheme for extracting object model from manipulation. The knowledge of the robot's arm motion gives us two precious information: 1) it enables us to segment the object from the rest of the scene; and 2) it allows to track object features in a robust manner. In combination with the visually induced grasping reflex presented in [2], this allows for an



(a) (b) (c) (d)

Fig. 4. Birth of an object (a)-(b) top:2D projection of the accumulated 3D representation and newly introduced primitives, bottom:accumulated 3D representation. (c) newly introduced and accumulated primitives in detailed. Note that, the primitives that are not updated are red and the ones that have low confidence are grey (d) final accumulated 3D representation from two different poses.

Fig. 5. Objects and their related accumulated representation.

exploratory behaviour where the robot attempts to grasp parts of its environment, examine all successfully grasped shapes and learns their 3D model and by this becomes an important submodule of the cognitive system discussed in [5].

Acknowledgement: This paper has been supported by the EU-Project PACOplus (2006-2010).

References

1. C. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3D objects. In *IEEE International Conference on Robotics and Automation*, pages 1890–1896, Detroit, Michigan, May 1999.
2. J. Sommerfeld D. Aarno, D. Kragic, N. Pugeault, S. Kalkan, F. Wörgötter, D. Kraft, and N. Krüger. Early reactive grasping with second order 3d feature relations. *IEEE Conference on Robotics and Automation (submitted)*, 2007. submitted.
3. James H. Elder. Are edges incomplete ? *International Journal of Computer Vision*, 34:97–122, 1999.
4. P. Fitzpatrick and G. Metta. Grounding Vision Through Experimental Manipulation. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 361:2165 – 2185, 2003.
5. Ch. Geib, K. Mourao, R. Petrick, N. Pugeault, M. Steedman, N. Krüger, and F. Wörgötter. Object action complexes as an interface for planning and robot control. *Workshop 'Toward Cognitive Humanoid Robots' at IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, 2006.
6. R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2000.
7. N. Krüger, M. Van Hulle, and F. Wörgötter. Ecovision: Challenges in early-cognitive vision. *International Journal of Computer Vision*, accepted.
8. N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence & the Simulation of Behavious, AISB Journal*, 1(5):417–427, 2004.
9. D.G. Lowe. Three–dimensional object recognition from single two images. *Artificial Intelligence*, 31(3):355–395, 1987.
10. A.T. Miller, S. Knoop, H.I. Christensen, and P.K. Allen. Automatic grasp planning using shape primitives. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1824–1829, 2003.
11. N. Pugeault, F. Wörgötter, and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proc. IEEE Workshop on Perceptual Organization in Computer Vision (in conjunction with CVPR'06)*, 2006.
12. B. Rosenhahn, O. Granert, and G. Sommer. Monocular pose estimation of kinematic chains. In L. Dorst, C. Doran, and J. Lasenby, editors, *Applied Geometric Algebras for Computer Science and Engineering*, pages 373–383. Birkhäuser Verlag, 2001.

Rigid Body Motion in an Early Cognitive Vision Framework

IEEE 5th Chapter Conference on Advances in Cybernetics Systems 2006

Nicolas Pugeault
University of Edinburgh, UK
npugeaul@inf.ed.ac.uk

Florentin Wörgötter
University of Göttingen, Germany
worgott@chaos.gwdg.de

Norbert Krüger
Sydansk University, Denmark
nk@imi.aau.dk

Abstract – *In this work we estimate the ego–motion from stereo sequences. We approach this problem in an early cognitive vision framework, i.e., we utilise structural interdependencies in visual data by recurrent predictive processes. More specifically we make use of rich and condensed local image descriptors (so called 'multi-modal primitives') to find correspondence sets with a large proportion of correct correspondences that become further improved by perceptual grouping.*

We use those correspondence sets to compute the ego–motion in a variety of scenes of different complexity and we show that our motion estimates are reliable and precise enough to be used as a predictor in our early cognitive vision framework.

Keywords: RBM, estimation, cognitive vision, multi-modal, feature-based, ego-motion.

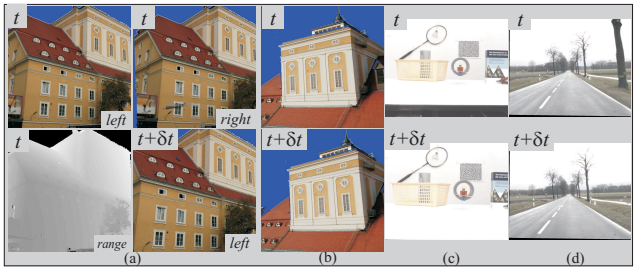


Figure 1: Two frames, one before the motion and one afterwards for the four sequences used in this paper. For sequences (a) and (b) we have depth values obtained from a range scanner and an exact ground truth for the camera motion (see picture (a) bottom-left). Sequence (c) was recorded indoor under controlled conditions, and the motion was measured during the recording. Finally, sequence (d) was recorded outdoor.

1 Introduction

Motion estimation is an important but complex problem in computer vision (e.g. [1, 3]). Furthermore, an accurate estimation of ego–motion is critical for active systems, for the purpose of map building (SLAM), obstacle avoidance, path planning, *etc.* Three sub-problems need to be addressed in a motion estimation algorithm:

Correspondence problem: In order to constrain the motion, we need to identify correspondences between stereo images recorded before and after the motion. Hence this correspondence problem is: finding correspondences over stereo and over time.

Mathematical formalisation: Different frameworks (e.g. matrices, quaternions, dual quaternions, *etc.* See [9] for a comparison) have been developed to formalise the Rigid Body Motion (RBM). The correspondences yield different constraint equations depending on the mathematical framework used.

Dealing with wrong correspondences: Since not all correspondences will be correct, the use of statistical methods is required to eliminate the erroneous correspondences. A prominent example is RANSAC [2].

These problems are deeply intertwined. For example, the mathematical formalisation of the constraints depends on the entities that are used. Also, what are 'good entities' is relative to the context: e.g. corners are less frequent than line segments but lead to stronger constraints. Note that, at one end of the spectrum, some try to extract the motion directly from the signal without extracting any features — see [4]. For an overview of the feature–based techniques we refer to [15].

In this paper, we estimate the ego–motion from calibrated stereo sequences of complex scenes in an early cognitive vision framework (see [16, 5]). By early cognitive vision we mean a stage of visual processing that comes after the processing of the visual signal into different modalities such as orientation, colour, optic flow and stereo disparity (the early vision stage). In early cognitive vision, these different modalities become interconnected by recurrent loops realising predictions between visual events and, thus, lead to reliable and structured scene representations. We will show that the motion estimation problem can be efficiently addressed within such a framework. Furthermore, the computed motion itself is the basis for strong predictive mechanisms that can be used to disambiguate scene representations.

2 Early Cognitive Framework

The mathematical framework we apply is based on the pose estimation algorithm developed by Rosenhahn et al. [14], which requires correspondences between 3D and 2D

entities. In our case this translates into two correspondence problems of rather different nature: The stereo correspondence problem and the temporal correspondence problem (see figure 4). When looking for correspondences, we face three sub–problems:

Projective transformation of visual entities: Differences in projection may change drastically the local image structure, and make it difficult (or even impossible in cases of occlusion) to find the correspondence to a primitive in the second image.

Ambiguity of visual entities: Due to the natural redundancy of local signals in images, the structure of a local image patch may be very similar to other areas in this image.

Local signal noise: Noise in the image signal may result in rather different visual entities being extracted from corresponding locations.

Because of these three problems, simple pixel intensity (or colour) information is not well suited for addressing this correspondence problem. Here, we will use a novel image representation, developed by [6], which describes the image in terms of primitives (as illustrated in figure 2).

2.1 Visual Primitives

The image processing used in this paper is based on multi–modal visual primitives [6, 11]. Primitives are extracted sparsely at points of interest in the image (in this case contours), and encode the value of different visual operators: position $\boldsymbol{m}$, orientation θ, phase ω, colour $\boldsymbol{c}$ and local optical flow $\boldsymbol{f}$, hereby referred to as *modalities* (see figure 2 b) and c)). Consequently a primitive is described by the following *multi–modal* vector:

$$\boldsymbol{\pi} = (\boldsymbol{m}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T \tag{1}$$

where ρ is the size of the image patch that was used to generate the primitive. By encoding a local image patch by such a multi–modal vector we achieve a condensation of information by a factor of 97% (see [8]).

2.2 Stereopsis using 2D–primitives

Classical stereopsis allows reconstructing a 3D point from two corresponding stereo points. This problem has been extensively studied in the computer vision literature (see, e.g., [1, 3]). The stereo–matching of visual primitives was previously studied in [10, 8], and we make use the same multi–modal matching criterion in the present paper:

$$d_m(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j) = \sum_m w_m d_m(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j) \tag{2}$$

where w_m is the relative weighting of the modality m, with $\sum_m w_m = 1$ In figure 3 the performance of using the distance in each modality is compared with the multi–modal distance.

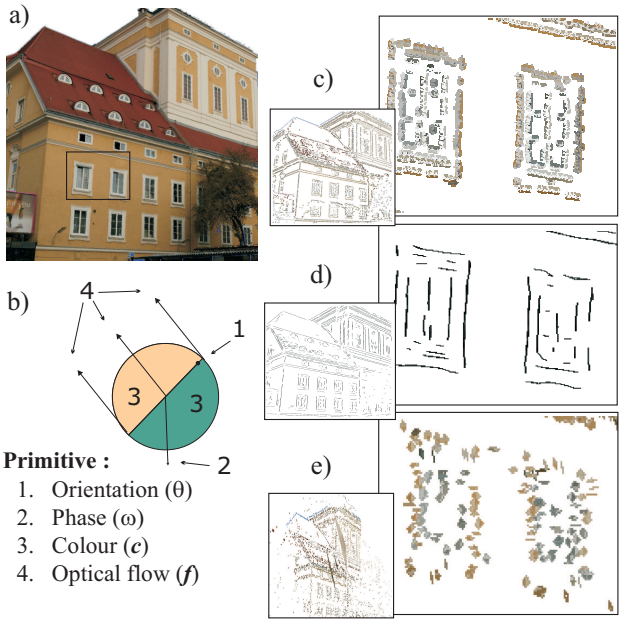


Figure 2: Illustration of the primitive extraction process. (a) shows one image of the object to be grasped; the symbolic representation of the image primitives is illustrated in (b). The primitives extracted from the image (a) are shown in (c), and the groups obtained by perceptual grouping in (d). Finally, (e) shows the 3D hypotheses reconstructed from a stereo–pair of images of the object.

Moreover, as primitives contain more information than a mere point, a stereo–pair of corresponding primitives allows for the reconstruction of a more complex kind of entity, thereafter called 3D–primitives $\boldsymbol{\Pi}$, such that:

$$\boldsymbol{\Pi} = (\boldsymbol{M}, \boldsymbol{\Theta}, \Omega, \boldsymbol{C})^T \tag{3}$$

where $\boldsymbol{M}$ is the location in space, $\boldsymbol{\Theta}$ is the 3D orientation of the edge, Ω is the phase across this edge, and $\boldsymbol{C}$ holds the colour information for this edge.

2.3 Perceptual grouping for Stereopsis

One additional advantage of visual primitives is the semantic information they carry. In [11] it has been shown that perceptual grouping information could be used successfully to disambiguate such a stereo–matching process.

Isolated primitives are likely to be unreliable: As primitives are redundant along contours, conversely an isolated primitive cannot describe any contour, and is likely to be an artifact of the primitive extraction. Hence isolated primitives can (and should) be neglected.

Stereo consistency over groups: If a set of primitives is part of a contour in the first image, their *correct correspondences* in the second image also form a contour.

Figure 3 (taken from [11]) shows the performance of the stereo–matching on the sequences presented in figure 1, when using different matching criterion. The figure ROC curves of the stereo–matching process when using the similarity in each of the individual modalities, the multi–modal similarity and a combination of this multi–modal criterion

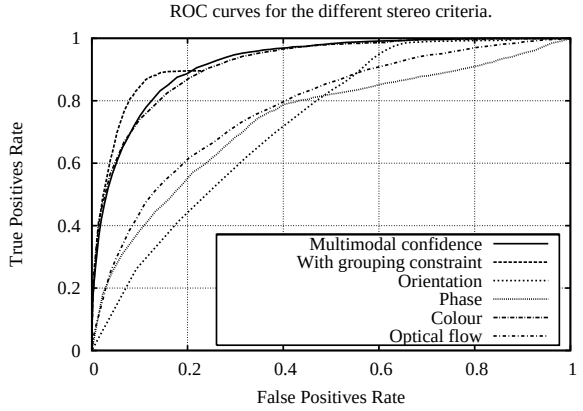


Figure 3: ROC curves illustrating the reliability of the stereo matches obtained. The first curve shown is for the multi–modal similarity metric defined in [10]. The second shows the improvement when enforcing an additional grouping constraint. The four last curves represent the performance when using each of the modality only for stereo matching — see [11] for a discussion of these curves.

with the aforementioned perceptual grouping constraints. A better matching is characterised by larger area under the curve. This figure shows that using the combined criterion improves significantly the reliability of the stereo–matches that were found. Those results are discussed in [11].

3 Ego–motion Estimation

The mathematical framework we apply is based on the pose estimation algorithm developed by Rosenhahn et al. [12, 14], that makes use of correspondences over time between 3D and 2D entities. This is illustrated in figure 4, where the left (respectively right) column shows the images obtained from the left (resp. right) camera, while the two rows capture two different instants in time. The estimation of the 3D motion of the camera between those two instants translates into two correspondence problems of rather different nature:

Stereo matching: Considering a primitive $\boldsymbol{\pi}_{l,t}$ in the left image, we want to find the corresponding primitive $\boldsymbol{\pi}_{r,t}$ in the right image, and use such correspondences to reconstruct 3D primitives $\mathbf{\Pi}$. This is illustrated by the upper row of figure 4, and has been addressed in the previous section.

Temporal matching: In order to estimate the motion, we need to find correspondences between the 3D–primitives reconstructed at time t, and the 2D–primitives extracted from the images at time $t+\delta t$ (bottom row of figure 4). The position of the 2D–primitive $\boldsymbol{\pi}_{l,t}$ and the optical flow thereof are used to predict the location of the correspondence $\boldsymbol{\pi}_{l,t+\delta t}$ at time $t+\delta t$. The multi–modal similarity between these two primitives is used to select the most likely correspondence in a neighbourhood of the predictor, or to discard the prediction altogether if no suitable match is found.

Stereo–temporal consistency check: We additionally constrain the matches found in the left and right images of any given 3D–primitive such that they comply with the epipolar geometry of the stereo set–up. In other words, we validate a stereo–pair of temporal matches only if the correspondence in the right image lies sufficiently close by the epipolar line defined by the correspondence in the left image. (see figure 4, bottom right).

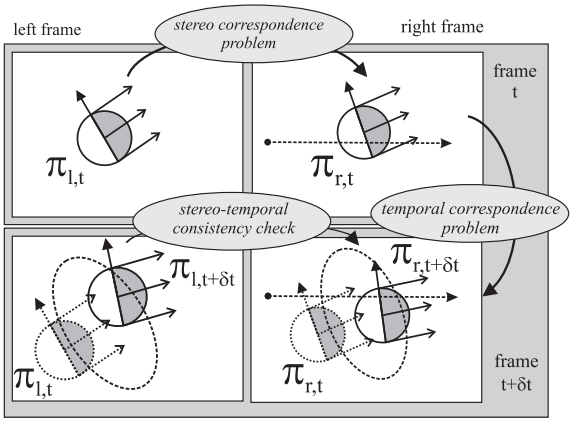


Figure 4: The two kinds of image feature correspondences required to estimate the RBM (see text).

3.1 Finding 3D–2D correspondences over time

A large part of the problem is to find accurate correspondences between (correct) 3D–primitives reconstructed before the motion (see section 2.2) on the one hand, and the image primitives extracted from images after the motion (see figure 4) on the other hand. In this section we propose a simple scheme to match reliable 3D–primitives with the 2D–primitives extracted at a later stage. We will call $\boldsymbol{\pi}_i^{F,t}$ the i^{th} image primitive extracted at time t, from the frame $F \in \{L, R\}$ (left or right). The first cue we can use for this matching is the position of the primitive $\boldsymbol{\pi}_{i,t}$ and the 2D optical flow thereof. This gives us an *a priori* estimate for the location $m_{i,t+\delta t}$ of the same primitive at a later stage $t+\delta t$ (see figure 4).

All primitives extracted at time $t+\delta t$ in the vicinity of this predicted location $m_{i,t+\delta t}$ are therefore considered potential correspondences of $\boldsymbol{\pi}_{i,t}$ at time $t+\delta t$. The neighbourhood has a radius of $\gamma\delta t$ times the size ρ of the receptive field of the primitives ($\gamma\rho\delta t$). γ is a constant that steers the selectivity of the matching process. Then out of all potential matches $\boldsymbol{\pi}_{j,t+\delta t}$ the one most similar to $\boldsymbol{\pi}_{i,t}$ is chosen. If all potential matches prove to be too dissimilar to the original primitive, they are all discarded. Note that here we mean by similarity the equation (2).

If we consider on the one hand a primitive $\boldsymbol{\pi}_{i,t}$ located at position $m_{i,t}$ at time t, with an optical flow vector of $\boldsymbol{f}_{i,t}$, on the other hand a primitive $\boldsymbol{\pi}_{j,t+\delta t}$ located at position $m_{j,t+\delta t}$ at a later time $t+\delta t$, and write $d_E(\boldsymbol{\pi}_{i,t}, \boldsymbol{\pi}_{j,t+\delta t})$ the Euclidean distance between those primitives and $d_m(\boldsymbol{\pi}_{i,t}, \boldsymbol{\pi}_{j,t+\delta t})$ the multi–modal distance between them; then the condition for the primitive $\boldsymbol{\pi}_{j,t+\delta t}$ to be considered as a correspondence of $\boldsymbol{\pi}_{i,t}$ can be sum-

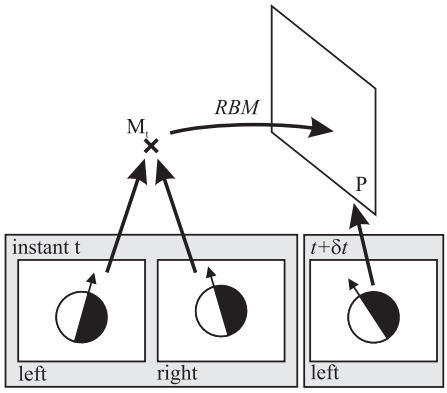


Figure 5: Illustration of the 3D–point / 2D–line constraint. A stereo–pair of primitives at time t yields a position $\boldsymbol{M}_t$ in space. Then, if we have a matching primitive $\boldsymbol{\pi}_{t+\delta t}$ in one image at time $t+\delta t$, we know that the 3D–point $\boldsymbol{M}_t$ has moved such that $\boldsymbol{M}_{t+\delta t}$ lies on the 3D plane P that projects into the line that is defined by the primitive $\boldsymbol{\pi}_{t+\delta t}$.

marised as follows:

$$\left\{ \begin{array}{rcl} d_E(m_{j,t+\delta t}, m_{i,t} + \boldsymbol{f}_{i,t}) & < & \gamma\rho\delta t \\ d_m(\boldsymbol{\pi}_{i,t}, \boldsymbol{\pi}_{j,t+\delta t}) & < & \varepsilon \end{array} \right. \quad (4)$$

where ε is a quantity small enough to only allow correspondences between fairly similar primitives. If several $\boldsymbol{\pi}_{j,t+\delta t}$ are satisfying those constraints, the one minimising $d_m(\boldsymbol{\pi}_{j,t+\delta t}, \boldsymbol{\pi}_{i,t})$ is chosen (see also figure 4).

3.2 3D–point / 2D–line correspondences

The pose estimation algorithm proposed is based on the method proposed by Rosenhahn *et al.* [12], that makes use of an exponential formulation of the motion (twist). In this work, we will focus on estimating the RBM from sets of 3D–point / 2D–line correspondences, as the ones we defined above. We consider the 3D–point $\boldsymbol{M}_t$ at time t, obtained applying stereo to a pair of images $\mathcal{I}_{L,t}$ and $\mathcal{I}_{R,t}$. Then we constrain the estimate of the motion in space of this point over time span δt such that it re-projects onto primitive $\boldsymbol{\pi}_i^{L,t+\delta t}$ in image $\mathcal{I}_{L,t+\delta t}$ at instant $t+\delta t$. As the possible origins of the primitive $\boldsymbol{\pi}_{i,t+\delta t}^{L}$ are contained in a plane in space P, the previous statement equates to constraining the position of the points after motion $\boldsymbol{M}_{t+\delta t}$ to the plane P — see figure 5.

If we now write the 3D–point $\boldsymbol{M}_t = (m_x, m_y, m_z)$, and define P by its normal vector $(n_x, n_y, n_z)^T$ and its Hesse distance to the origin h, we can rewrite this constraint as follows (see [13, 7]):

$$\begin{pmatrix} n_x \\ n_y \\ n_z \\ -n_z m_y - n_y m_z \\ -n_x m_z - n_z m_x \\ -n_y m_x - n_x m_y \end{pmatrix}^T \cdot \alpha \begin{pmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega z \end{pmatrix} = D \quad (5)$$

$$D = -h - n_x m_x - n_y m_y - n_z m_z \quad (6)$$

In this formula, $\boldsymbol{v} = (v_x, v_y, v_z)$, $\boldsymbol{w} = (w_x, w_y, w_z)^T$ is the twist representation of the unknown motion (see [12]) while α is a parameter relevant for the iterative solution of the system of linear equations.

A Rigid Body Motion contains 6 degrees of freedom (DOF), hence in order to solve this system of linear equation we need to draw at least 6 constraints from the pool of correspondences. As we intend to match a 3D point in the left and right images, each of those stereo–correspondence yield two constraints and so we need a set of only 3 correspondences to compute a RBM. Obviously, as we expect to get a pool $\mathcal{C}$ of at least several hundred of correspondences, the problem is largely over-constrained.

On the other hand, we face the problem that a certain quantity of those correspondences is expected to be erroneous. Moreover, for the reasons stated earlier, a certain inaccuracy in the reconstruction of the 3D–points is also to be expected. The problem of motion estimation is then to select, out of this large pool $\mathcal{C}$, a set of correspondences $\mathcal{S} \subset \mathcal{C}$, which generates an accurate ego–motion estimate.

In the following we will propose an estimator of the accuracy of the computed motion that can be computed on-line, even if the real motion is not known (section 3.3). We then discuss three strategies to select suitable subsets $\mathcal{S} \subset \mathcal{C}$ for motion estimation in section 3.4.

3.3 On–line estimation of the RBM quality

For each set $\mathcal{S} \subset \mathcal{C}$ of correspondences, a rigid body motion $RBM_{\mathcal{S}}$ can be computed. If erroneous correspondences are included into $\mathcal{S}$ then the computed $RBM_{\mathcal{S}}$ is bound to be inaccurate too. In order to choose $\mathcal{S}$ adequately, we need a measure of the quality of the computed RBM, and thus of the set which generated it.

We propose to estimate the quality of a computed RBM using the reliability of the predictions generated thereof. For each correspondence in the pool $\mathcal{C}$, we consider the associated 3D–primitive $\boldsymbol{\Pi}_{i,t}$ that was reconstructed at time t, and predict its position $\hat{\boldsymbol{\Pi}}_{i,t+\delta t}$ at time $t+\delta t$, according to the assumed motion $RBM_{\mathcal{S}}$; then those 3D predictions are re–projected onto both image planes and compared with the actual correspondences, from the 2D–primitives extracted at time $t+\delta t$.

We compute the *deviation* of a given correspondence from the RBM as the sum of the normal distances between on the one hand the prediction reprojected on both frames $\hat{\boldsymbol{\pi}}_i^{L,t+\delta t}$ and $\hat{\boldsymbol{\pi}}_i^{R,t+\delta t}$ (that are predictions derived from primitives extracted at time t) and on the other hand the actual correspondences $\boldsymbol{\pi}_j^{L,t+\delta t}$ and $\boldsymbol{\pi}_j^{R,t+\delta t}$ (that are primitives which have been extracted from the image at time $t+\delta t$). Here normal distance is understood as, e.g., the distance between the position of $\hat{\boldsymbol{\pi}}_i^{L,t+\delta t}$ and the line defined by the orientation of $\boldsymbol{\pi}_j^{L,t+\delta t}$. Effectively, the deviation is the difference between the predicted image representation and the extracted one. We define the mean deviation $\langle\Delta\rangle$ for a given $RBM_{\mathcal{S}}$ (estimated from a a subset $\mathcal{S}$) as the average of the deviations of all the correspondences of the

pool $\mathcal{C}$ for this motion:

$$\langle \Delta \rangle = \sum_{i \in P} \frac{d_n(\boldsymbol{\pi}_i^L, \boldsymbol{\pi}_j^L) + d_n(\boldsymbol{\pi}_i^R, \boldsymbol{\pi}_j^R)}{2\ \#\mathcal{C}} \tag{7}$$

Note that in this formulation erroneous correspondences will lead to data points with very large deviation, and therefore influence drastically the mean deviation $\langle \Delta \rangle$. The reliability of this measure could be improved by disregarding the data points with large deviations, as in, e.g., RANSAC (see [2]).

3.4 Three Strategies to choose suitable Subsets

The rich descriptors embedded in the early cognitive vision framework lead to a pool $\mathcal{C}$ containing a rather small number of wrong correspondences. We will discuss in the following three strategies that were employed to select optimal sets $\mathcal{S}$ of correspondences from $\mathcal{C}$: random sets, growing sets and RANSAC.

Note that, in order to obtain a more reliable estimate of the quality of the RBM selected, we will henceforth split $\mathcal{C}$ into two subsets: the generation set $\mathcal{C}_G$ and the validation set $\mathcal{C}_V$. In the following, correspondences are drawn from $\mathcal{C}_G$ when computing the motion and estimating the mean deviation to select the best motion out of several candidates; correspondences were drawn from the validation subset $\mathcal{C}_V$ when computing the $\langle \Delta_V \rangle$ of the elected motion estimate. Therefore the deviation value shown in in figure 6 is $\langle \Delta_V \rangle$, drawn from the correspondences in the *validation* set $\mathcal{C}_V$.

Random sets: The RBM estimation has been processed for 100 different sets each containing $n \in [3, 20]$ correspondences picked randomly from the generation pool. The performance of the set resulting in the lowest deviation $\langle \Delta_G \rangle$ of the generation set $\mathcal{P}_G$ for each set size is drawn in the figure 6.

Dynamic growing of a set of correspondences: The obvious drawback of extending the size of a randomly chosen set of correspondences is that the likelihood to include false correspondences in the set increases exponentially with the size of the set. We propose to generate larger sets of correspondences more reliably, by growing them from smaller — and so more reliable — sets. We proceed as follows: 1) we generate randomly a population of 100 sets $S_{n,i}$ of a small size n — as described earlier; 2) the set S_n minimising the deviation $\langle \Delta_G \rangle$ (over the generation set $\mathcal{C}_G$) is chosen, as before; 3) we create a new population $\mathcal{S}_{n+1,i}$ of sets of size n+1, such as $\mathcal{S}_n \subset \mathcal{S}_{n+1,j}$; 4) back to step 2 until $\mathcal{S}$ reached a certain size m.

Random Sample Consensus (RANSAC): RANSAC is a paradigm proposed in [2] to select efficient sets of constraints from a pool of unreliable data points. The problem of incorrect data points is that they result in very large deviations, even for the correct motion, and so make the average deviation a very noisy measure (as the impact of erroneous data points is several orders of magnitude stronger than the impact of the correct points). RANSAC proposes to address this problem by only considering data-points which have a deviation within a certain tolerance.

4 Results on Ego–motion Estimation

We illustrate the performance of the system using a variety of sequences, from very controlled scenes (figures 1(a) and (b)) for which the actual motion is known, to more realistic scenarios (figure 1(c) and (d)). The ground truth of the motion is only known accurately for sequences (a) and (b), respectively a pure translation of 2 meters along the positive z axis, and a pure rotation of 0.2 radians around the vertical y axis (to the left). For sequence (c) we have measured the motion when recording being 56.5mm. along the positive z axis. Each strategy has been computed over 100 trials, and the estimate leading to the smallest deviation is selected.

As the motion is computed using correspondences from 2D images, its accuracy is limited by the coarseness of the pixel sampling (although the primitive extraction offer some measure of sub–pixel accuracy) and depends on the projection operated by the cameras. In short, pixel sampling creates an inaccuracy in the 3D position estimations which is proportional to the distance of the 3D–point to the camera — see [1] for a mathematical demonstration. Effectively, we will use the mean deviation $\langle \Delta \rangle$, that was introduced above (equation 7), to estimate how accurately the computed 3D motion allows us to predict the motion flow of 2D–primitives.

In figure 6, the first row of plots refers to the sequence (a), for which motion was known, and shows $\langle \Delta_V \rangle$, as well as the translation error in millimetres and the rotation angle error in radians. Those plots show the performance obtained when using each of the three strategies proposed earlier, and depending on the size of the random set of correspondences $\mathcal{S}$ the motion were computed from. The error decreases for any size of $\mathcal{S}$ larger than 3 pairs of correspondences, and then remains stable. In this sequence the deviation is consistently below 5 pixels (which is less than the size of a primitive). The translation estimation error is of approximately 0.2mm. which creates an uncertainty of 10% of the translation. The rotation component (which is null in this sequence) suffers an error of only 0.2 radians.

The second row shows $\langle \Delta_V \rangle$ against the size of $\mathcal{S}$ for the three other sequences. For sequence (b) the deviation falls to 5 pixels for sets of 4 correspondence pairs for RANSAC, 8 correspondence pairs for the growing sets and 14 for the random sets. Sequence (c) is significantly more difficult, leading to more outliers in the correspondence pool, thus $\langle \Delta_V \rangle$ converges to approximately 8 pixels instead of 5 for (a) and (b). Here again, there is no significant difference between the three strategies when choosing larger $\mathcal{S}$. Finally sequence (d) is proves more difficult, leading to some instability in the motion estimation obtained using the random and RANSAC method (from 7 to 15 pixels of deviation), while the growing method consistently yields approximately 7 pixels of deviation. For comparison, when com-

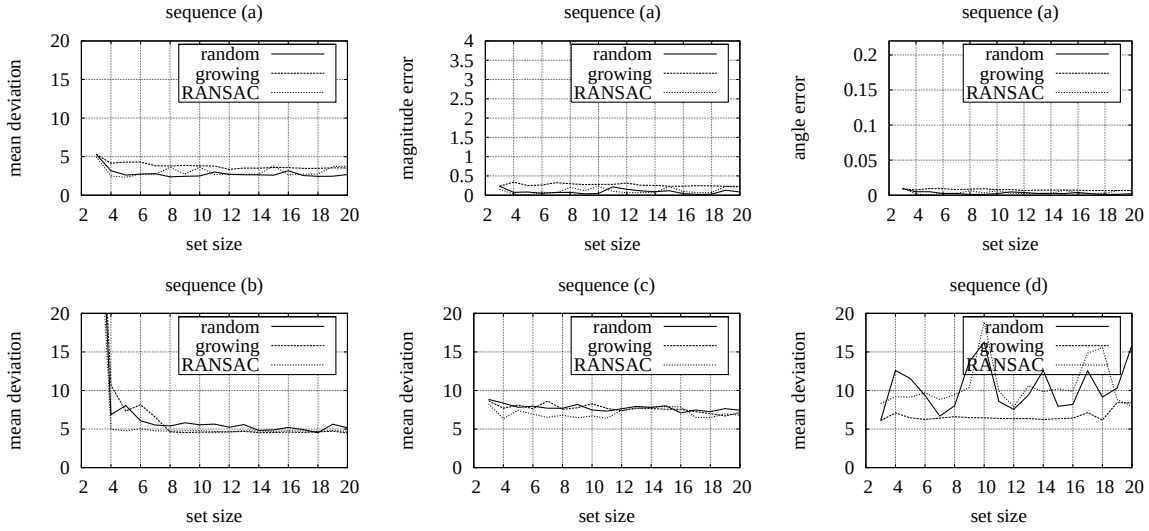


Figure 6: Comparative results for random sets, RANSAC, growing for sequences (a), (b), (c) and (d) — see figure 1. The graph shows $\langle \Delta_V \rangle$, the translation magnitude and the rotation angle error for the sequence (a) for which the true motion was known. For sequences (b), (c) and (d) only $\langle \Delta_V \rangle$ is shown.

puting the RBM using sets of 20 hand–picked the average mean deviation was close to 5 pixels.

5 Conclusion

We have shown that an early–cognitive approach by integrating information across modalities (colour, optical flow, *etc.*) and visual processes (perceptual grouping and stereopsis) enables us to address the correspondence problem with good reliability. This allows the reliable reconstruction of a 3D description of the scene, and a robust matching of image features over time. The relatively low number of outliers enables us to compute a robust estimation of the ego–motion in a variety of sequence, even without resorting to a sophisticated statistical method to select the correspondence set.

The reliability of this correspondence finding is shown to be sufficient to allow for a robust ego–motion estimation, without prior knowledge about the scene or the motion. Furthermore the deviations obtained when using this ego–motion for making predictions are significantly lower than the size of a primitive (13 pixels), allowing for an unambiguous tracking of these low level features.

This robust tracking of the low level visual feature has been used successfully for outlier removal, noise correction and improvement over time of early hypotheses. We believe that such feedback loops between higher level information and lower level image descriptors are essential in the design of robust and versatile cybernetics systems solely based on visual input.

Acknowledgement: We thank the company Riegl for the images with known ground truth used for sequence (a) and (b). The work described in this paper was part of the European project ECOVISION.

References

[1] Olivier Faugeras. *Three–Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.

[2] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981.

[3] R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[4] Michal Irani and P. Anandan. About direct methods. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 267–277, London, UK, 2000. Springer-Verlag.

[5] N. Krüger, M. Van Hulle, and F. Wörgötter. Ecovision: Challenges in early-cognitive vision. *International Journal of Computer Vision*, submitted.

[6] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence & the Simulation of Behavious*, 1(5):417–427, 2004.

[7] N. Krüger and F. Wörgötter. Statistical and deterministic regularities: Utilisation of motion and grouping in biological and artificial visual systems. *Advances in Imaging and Electron Physics*, 131:81–146, 2004.

[8] Norbert Krüger and Michael Felsberg. An explicit and compact coding of geometric and structural information applied to stereo matching. *Pattern Recognition Letters*, 25(8), 2004.

[9] A. Lorusso, D. W. Eggert, and R. B. Fisher. A comparison of four algorithms for estimating 3-d rigid transformations. In *Proceedings of the British Machine Vision Conference*, 1995.

[10] N. Pugeault and N. Krüger. Multi–modal matching applied to stereo. In *Proceedings of the British Machine Vision Conference 2003*, 2003.

[11] Nicolas Pugeault, Florentin Wörgötter, and Norbert Krüger. Multi–modal scene reconstruction using perceptual grouping constraints. In *Proceedings of the 5th IEEE Workshop of Perceptual Organization in Computer Vision*, 2006.

[12] B. Rosenhahn. *Pose Estimation Revisited*. PhD thesis, Institut für Informatik und praktische Mathematik, Chrsitian–Albrechts–Universität Kiel, 2003.

[13] B. Rosenhahn, O. Granert, and G. Sommer. Monocular pose estimation of kinematic chains. In L. Dorst, C. Doran, and J. Lasenby, editors, *Applied Geometric Algebras for Computer Science and Engineering*, pages 373–383. Birkhäuser Verlag, 2001.

[14] B. Rosenhahn, N. Krüger, T. Rabsch, and G. Sommer. Automatic tracking with a novel pose estimation algorithm. *Robot Vision 2001*, 2001.

[15] Philip H. S. Torr and Andrew Zisserman. Feature based methods for structure and motion estimation. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 278–294, London, UK, 2000. Springer-Verlag.

[16] F. Wörgötter, N. Krüger, N. Pugeault, D. Calow, M. Lappe, K. Pauwels, M. Van Hulle, S. Tan, and A. Johnston. Early cognitive vision: Using gestalt-laws for task-dependent, active image-processing. *Natural Computing*, 3(3):293–321, 2004.

Robotics Group
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark

Technical Report no. 2007 – 1

Structured Visual Events

Nicolas Pugeault, Norbert Krüeger and Florentin Wörgötter

January 23, 2007

Title	Structured Visual Events
Author(s)	Nicolas Pugeault, Norbert Krüeger and Florentin Wörgötter
Publication History	

1 Introduction

The human visual system is efficient at grouping together visual information that belongs to the same objects, regardless of noise and ambiguity. Salient objects immediately 'pop out' of the visual environment. Gestalt psychologists suggested that this emergence of some coherent sub–parts of the scene is driven by a certain number of rules, also called *Gestalt Laws*. These laws stated that certain regularities lead the visual system to group together visual information that would otherwise be, from a local signal viewpoint, distinct. Such laws included, e.g., proximity, good continuation, similarity and symmetry. Striking demonstrations of such a bias in the human visual system exist in the form of so–called *visual illusions*: e.g. the Kanisza triangle, where an illusory triangle is strongly perceived.
There has been discussions that such laws might be originated by statistical properties in natural images. This was later demonstrated by [18, 8, 12]. In [7] a statistical approach was used to extract close contours. The statistical part was mainly concerned with the pairwise grouping of local edge pixels. [4] proposed a complementary statistical scheme to extract global groups from such information. We believe that such an approach can be extended to extract salient image structures without prior assumption on the scene witnessed or the objects that constitute it, and we propose to call those *Structured Visual Events (SVE)*. The primordial sort of structural SVE is a contour, and in its simplest form, the line. As discussed in [4], the likelihood for accidental alignment of edge pixels (or alternatively local edge–like features) is decreasing with the square of the size of the contour. Such SVE correspond to the Gestalt law of *Good Continuation*, and therefore we propose that more SVE could be inferred according to the other aforementioned laws.
In the present work we will consider the following regularities:

- Parallelism
- Coplanarity (in space, described in [15]).
- Similarity (co-colority, described in [15]).
- Good continuation (described in [25]).

All of these regularities are defined in 3D space, or alternatively across stereo in both images — see [15] for a detailed description.
We will propose a simple scheme to extract salient locations in the images, salient in the sense of a statistical oddity that is likely to correspond to an object in the scene. We will use in conjunction the above-mentioned relations to segment the visual world into Structured Visual Events and background. Note that the segmentation of visual scenes is a difficult problem, that found some satisfying solutions in the limited case of foreground/background segmentation, but that is otherwise unsolved. [9]

2 Visual primitives

Numerous feature detectors exist in the literature (see [23] for a review). Each feature based approach can be divided into an interest point detector (e.g. [13, 3]) and a descriptor describing a local patch of the image at this location, that can be based on histograms (e.g. [5, 23]), spatial frequency [17], local derivatives [14, 10, 1] steerable filters [11], or invariant moments ([22]). In [23] these different descriptors have been compared, showing a best performance for SIFT-like descriptors.
The primitives we will be using in this work are local, multi–modal edge descriptors that were introduced in [20]. In contrast to the above mentioned features these primitives focus on giving a semantically and geometrically meaningful description of the local image patch. The importance of such a semantic grounding of features for a general purpose vision front–end, and the relevance of edge–like structures for this purposes were discussed in [6].
The primitives are extracted sparsely at locations in the image that are the most likely to contain edges. This likelihood is computed using the intrinsic dimensionality measure proposed in [19]. The sparseness is assured using a classical winner take all operation, insuring that the generative patches

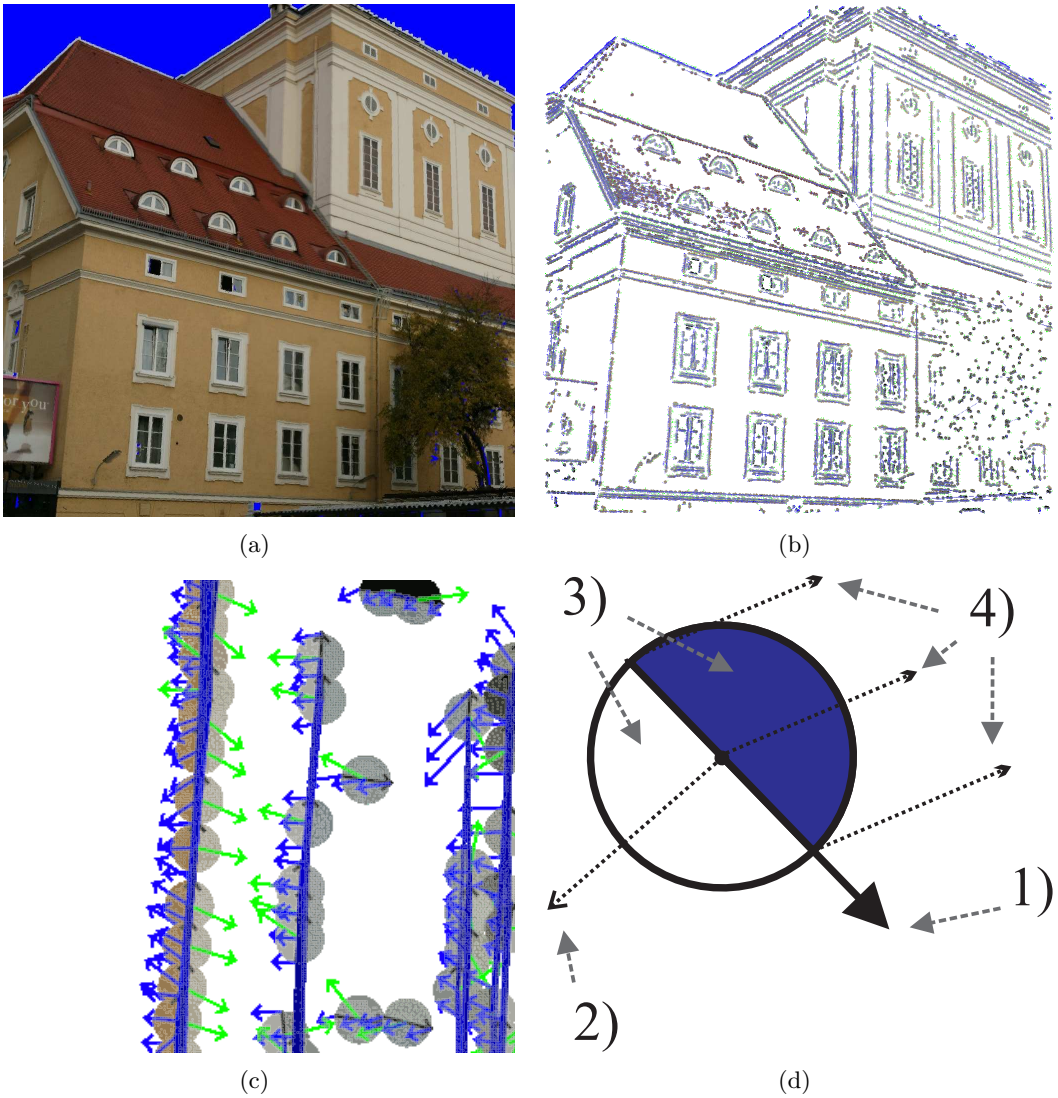
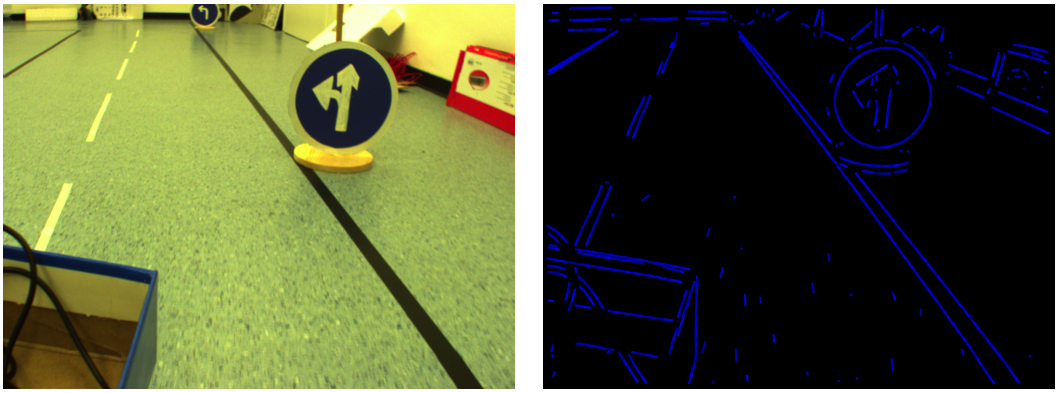


Figure 1: Illustration of the primitive extraction process from a video sequence. The figure shows in (a) one image from a video sequence on the right, then (b) the 2D–primitives extracted from this image, with a magnified version on (c). The blue lines between the primitives show the result of the perceptual grouping presented in [25] (d) describe the schematic representation of the 2D–primitives, where 1. shows the orientation of the primitive, 2. the phase, 3. the colour and 4. the optic flow.

of the primitives do not overlap (for details, see [21]). Each of the primitive encodes the image information contained by a local image patch. Multi–modal information is gathered from this image patch, including the position $\boldsymbol{m}$ of the centre of the patch, the orientation θ of the edge, the phase ω of the signal at this point, the colour $\boldsymbol{c}$ sampled over the image patch on both sides of the edge and the local optical flow $\boldsymbol{f}$. Consequently a local image patch is described by the following multi–modal vector:

$$\boldsymbol{\pi} = (\boldsymbol{m}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T, \tag{1}$$

that we will name *2D primitive* in the following. In this equation $\boldsymbol{m}$ refers to the position of the centre of the primitive in the image, θ is the orientation of the primitive, ω is the phase, $\boldsymbol{c}$ is the colour value, $\boldsymbol{f}$ is the local optic flow and ρ is the size of the primitive — see figure 1.
Note that these primitives are of lower dimensionality than, e.g., SIFT (10 vs. 128) and therefore suffer of a lesser distinctiveness. Nonetheless, as shown in [25] that they are distinctive enough for a reliable stereo matching if the epipolar geometry of the cameras is known. Furthermore, their semantic in terms of geometric and appearance based information allow for a good description of the scene content. It has been previously argued in [6] that edge pixels contain all important information



(a) image (b) collinear groups

Figure 2: Collinear groups extracted from a sample image.

in an image. As a consequence, the ensemble of all primitives extracted from an image describe the shapes present in this image.
Advantageously, the rich information carried by the 2D–primitives can be reconstructed in 3D, providing a more complete scene representation. Having geometrical meaning for the primitive allows to describe the relation between proximate primitives in terms of perceptual grouping.
In a stereo scenario a 3D–primitive $\mathbf{\Pi}$ can be computed from two corresponding 2D–primitives (see figure 1 and [25]): such that we have a projection relation:

$$\mathcal{P} : \mathbf{\Pi} \rightarrow \boldsymbol{\pi} \,. \tag{2}$$

A 3D–primitive $\boldsymbol{\pi}$ is described by the vector:

$$\mathbf{\Pi} = (\boldsymbol{M}, \Theta, \boldsymbol{\Omega}, \boldsymbol{C})^T, \tag{3}$$

where $\boldsymbol{M}$ is the location in space of the centre of the primitive, Θ is its orientation vector, $\boldsymbol{\Omega}$ is its phase and $\boldsymbol{C}$ holds the colour on both sides of the primitive.

3 Relations between primitives

In [15] a variety of relations that can be drawn between visual primitives were reviewed. In this paper we will focus on the following:

3.1 Collinearity

In [25] we proposed a simple scheme for grouping primitives that describe the same (smooth) contour of the scene. Herein we will assume that objects are delimited by piecewise smooth contours, joined by *junctions*. We will hereafter call *contour* these smooth sections.
Figure 2 shows the contours extracted by the grouping mechanism described in [25].

3.2 Proximity

The proximity relation is the fact that the two primitives, when re-projected onto both views are distant of less than a certain radius. The likelihood for a random occurrence of this relation is:

$$p(d_E(a,b) < \tau_E) = \frac{(\tau_E)^2}{\rho^2} p(\pi) \tag{4}$$

where $p(\pi)$ is the prior probability for the extraction of a primitive at a location.

$$p(\pi) = \frac{cr}{\#(\pi)\rho^2} \tag{5}$$

for a $c \times r$ image where $\#(\pi)$ primitives were extracted. Note that this two–dimensional definition of proxity is extended to 3D by enforcing that the re–projections on both image planes of the two 3D–primitives be proximate according to 2D definition.

3.3 Parallelism

We define the parallelism between two primitives as follows:

Definition 1. *Two primitives are said parallel if they share the same orientation.*

Therefore, collinearity is defined as follows:

$$||(a,b) = \text{acos}(\Theta_a \cdot \Theta_b) \tag{6}$$

If we consider that $||(a,b)$ is always between $[-\frac{\pi}{2}, +\frac{\pi}{2}]$ and if we consider as parallel all primitive pair (a,b) such that $||(a,b) < \tau_{\text{coll}}$, where τ_{coll} is the tolerance of the parallelism definition, then we have:

$$p_{\text{prior}}(\text{coll}(a,b) < \tau_{\text{coll}}) = \frac{\pi}{\tau_{\text{coll}}} \tag{7}$$

assuming normal distribution.[1]

3.4 Coplanarity

Coplanarity was defined in [15]. Note that the shape of circular contours tend to be inaccurately reconstructed (due to the nearly horizontal parts of the curve). Therefore the coplanarity relation is not very robust on circular structures.

3.5 Co–colority

We expect contours of the same surface to be co–colour. The co–colourity relation capture this prior knowledge about surfaces of the world. We will make use of this relation in conjunction with the parallelism and coplanarity relations to compensate for their relative statistical weakness. Co–colority is fully described in [15].

4 Relations between contours

As stated before, the relations between two primitives, taken individually are still statistically weak events. Moreover, we argued in [25] that contours and not primitives (that are merely local descriptors sampled from scene contours) should be used for scene description.
Therefore we will extend the relations mentioned in the previous section onto contours. In extending the definition to collinear groups, we want to generate *rarer*, and therefore more salient, events.

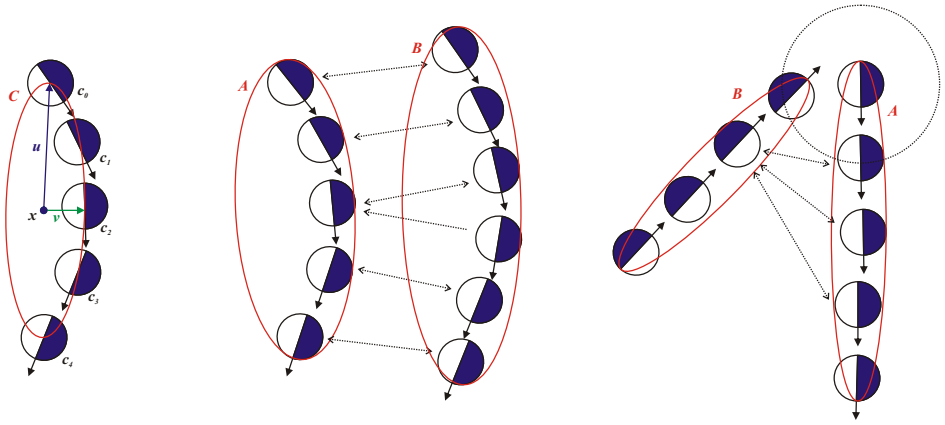
4.1 Symbolic representation of contours

From the pairwise good continuation relation proposed in [25] we propose to extract the whole contour by using a classical transitivity relation.

Definition 2. *If primitive A and B are linked, and B and C are linked, then A, B and C are part of the same contour.*

We describe the resulting contours with the four following measures:

[1] Given that horizontal and vertical edges are more common in natural scenes than other orientations, this assumption of a normal distribution do not hold. Nevertheless it is good enough as a working hypothesis. Having a proper model of the orientation co–occurrence would only serve to weight less horizontal collinear segments than other orientations, which would not serve any purpose for SVE extraction.



(a) collinearity (b) parallelism (c) coplanarity

Figure 3: Illustration of the primitive grouping scheme from the basic relations. (a) The contour A is extracted from primitives a_i that are all linked together by transitivity. (b) The two groups are parallel according to the global and local criteria. (c) The two groups A and B are coplanar. The dashed circle shows the proximity criterion.

- The center of mass ($\boldsymbol{x}$) of the positions of all primitives that belong to the contour.
- The main axis ($\boldsymbol{u}$): that is the first principal axis of the positions of the primitives that belong to the contour. Its norm is the elongation of the contour in this axis.
- The second axis ($\boldsymbol{v}$): that is the second principal axis. If the contour is perfectly rectilinear, then this axis is the null vector. Otherwise it is an indication of the curvature of the contour.
- The planarity (p) states how well the primitives are embedded into the plane formed by these two axes. In other word it is the elongation over the third principal axis.

And therefore a contour is described by the vector:

$$\boldsymbol{C} = (\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}, p) \tag{8}$$

Additionally a contour also contains the list of primitives that generated it, allowing for more complex comparison methods. In figure 3(a) the symbolic representation of the contours is shown. In the following we propose to use the contours, instead of the primitives themselves, to test for relations and evaluate saliency in a more powerful fashion.

4.2 Parallel contours

By extension we will consider that two contours A and B are parallel iff:

$$\begin{array}{l} \forall\, \boldsymbol{\Pi}_i \in A, \boldsymbol{\Pi}_j = \arg\min_{\boldsymbol{\Pi}_k \in B}(d_E(\boldsymbol{M}_i, \boldsymbol{M}_k)), \\ \left\{ \begin{array}{c} \text{——}(\boldsymbol{\Pi}_i, \boldsymbol{\Pi}_j) < \tau_{||} \\ |d_E(\boldsymbol{M}_i, \boldsymbol{M}_j) - d_E(A, B)| < \tau_E \end{array} \right. \end{array} \tag{9}$$

where $\boldsymbol{M}_i$ is the position of the primitive $\boldsymbol{\Pi}_i$, d_E is the Euclidian distance, $d_E(G_1, G_2)$ is the min distance between the two contours, and τ_E is the tolerance. Note that the second line is the global constraint for parallelism, whereas the first line is the local one. The first constraint statistically weakens with larger contours whereas the second one strengthens. Note that this interpretation of parallelism can capture curved contours that are equidistant in all points. This is illustrated in figure 3(b). In there the dashed arrows represent the primitives that are the closest between the two groups and the most parallel.

4.3 Coplanar contours

Building onto the definition of coplanarity between two primitives, we define that a primitive a is coplanar to a contour $B = (b_0, \ldots, b_n)$ iff

$$cop(a, B) \text{ iff } \frac{\#cop(a, b_i)}{\#B} < r \tag{10}$$

where r is a ratio, that we set to $r = 0.8$ for our experiments. A higher value will lead to a stricter definition whereas a lower one will find more cases of coplanarity.
Then we define the coplanarity between two contours $A = (a_0, \ldots, a_n)$ and $B = (b_0, \ldots, b_n)$ as

$$cop(A, B) \text{ iff } \begin{cases} \frac{\#cop(a_i \in A, B)}{\#A} < r \\ \frac{\#cop(b_j \in B, A)}{\#B} < r \\ \min_{a_i \in A, b_j \in B}(d_E(a_i, b_j)) < \tau_E \end{cases} \tag{11}$$

In other words, two groups are coplanar if a sufficient ratio of the primitives thereof are coplanar. Note that this can only occur for groups with a strong planarity. This is illustrated in figure 3(c), where the dashed circle shows the proximity criterion, and the dashed lines represent the two other criteria.

5 Results and discussion

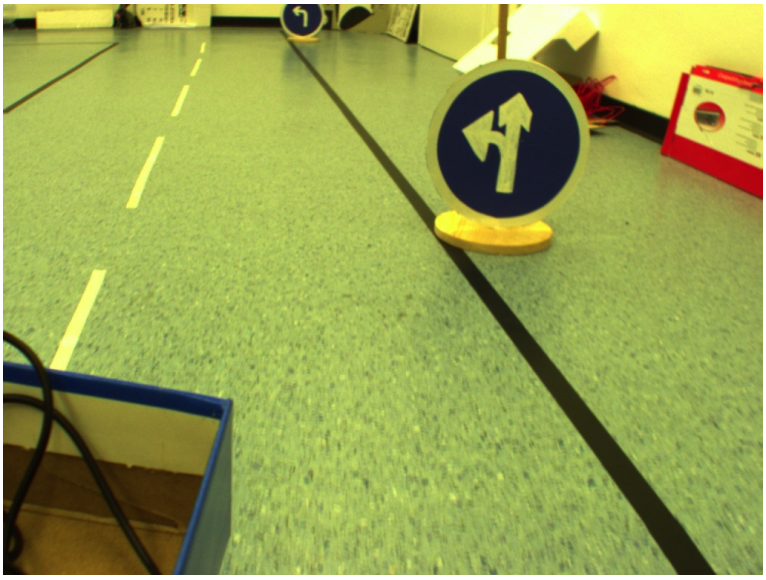
We applied these relations to some video simple sequences featuring some sample objects. For the purpose of these experiments we proceeded in extracting the SVEs in two steps:

1. extract 3D contours, as in [15].
2. Compute the relations between all contours.
3. merge all linked contour into one SVE.

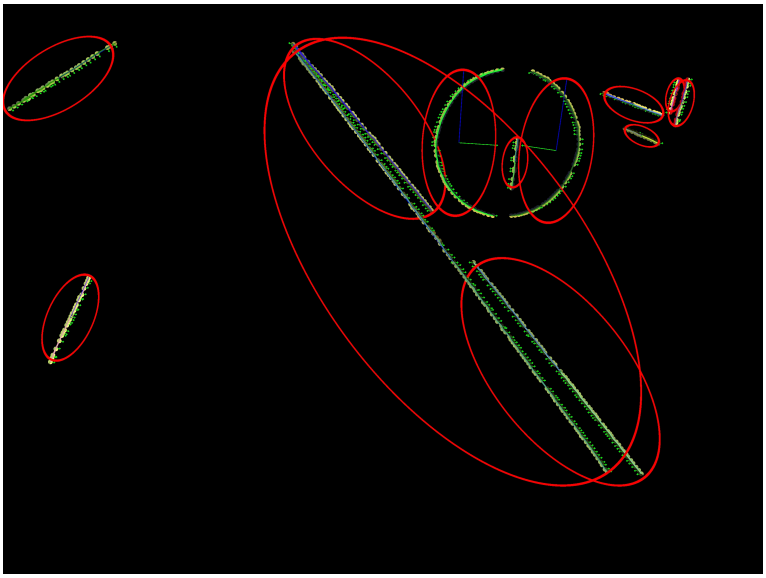
Note that this method is only used for experimentation purpose. In the future it would be preferable to keep the relational structure between all primitives instead of merging them all into one group.
We applied this method for two combinations of relations:

Parallelism + co–colourity: In this case we only considered the relation of parallelism. We also required that two primitives be co–colour in order to be considered as parallel. The results of this method applied to a driving scene are shown in figure 4. There we can see that the different parts of the white line are merged together. On the other hand, the two crescent–shaped parts of the traffic sign are left separate. Also the three lines on the ground, although parallel are not merged. This is due to the proximity constraint that we enforced in the definition of contour parallelism.

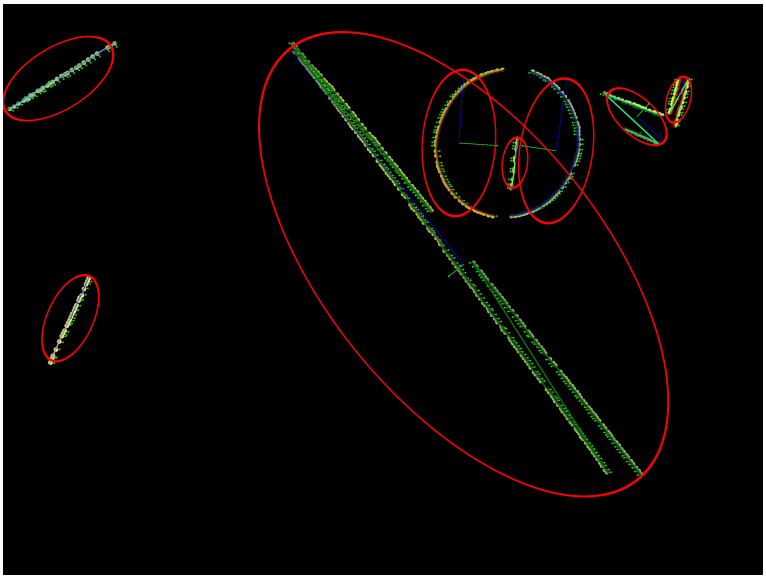
Coplanarity + co–colority For the second experiment we replaced the parallelism relation by the somewhat weaker coplanarity relation. Here again we required that the co–colourity be respected to consider two primitives as coplanar. The results applied to a sequence showing a traffic sign are shown in figure 5. Note that both sides of the support and both sides of the traffic signs a successfully grouped. The horizontal part of the traffic sign suffer from a reconstruction of low accuracy (because it is horizontal) and therefore the coplanarity is too weak to merge it. The results when applied to another scene featuring two mugs on a table are shown in 6. There we can see that the corners of the table are successfully merged. Here again the horizontal parts lead to problems: the horizontal border of the table is not grouped. Moreover because the reconstruction of the circular opening of the cups is inaccurate due to the horizontal (and curved) parts, some parts of the cups are found coplanar where they should not.



(a)



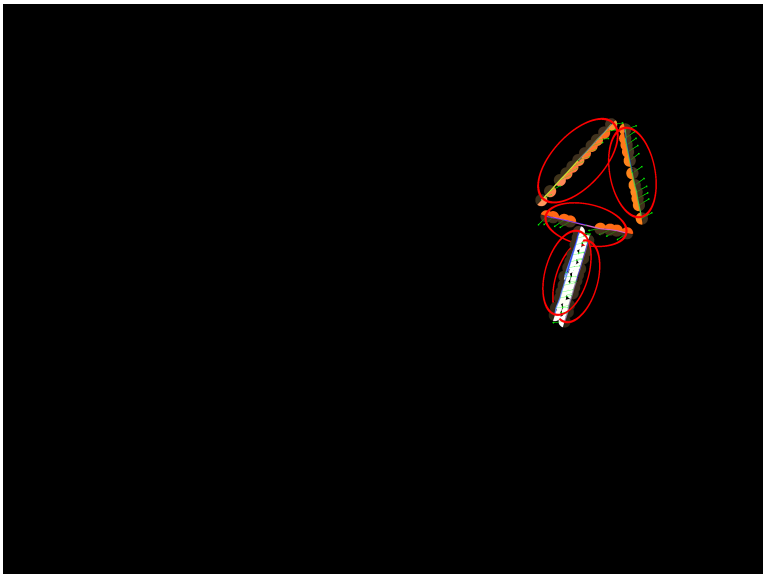
(b)



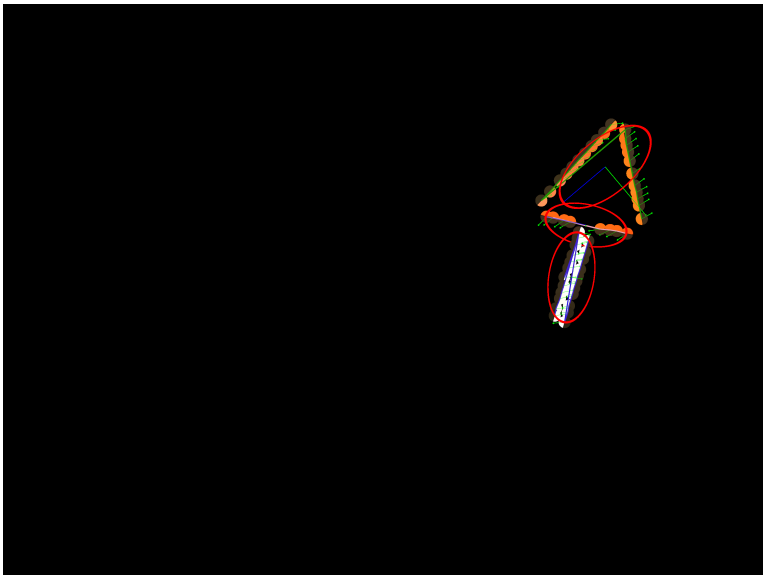
(c)

Figure 4: Example of the extraction of Visual Gestalts using good continuation (b) and parallelism + co–colourity (c) (the red ellipses show the Gestalts).

(a)

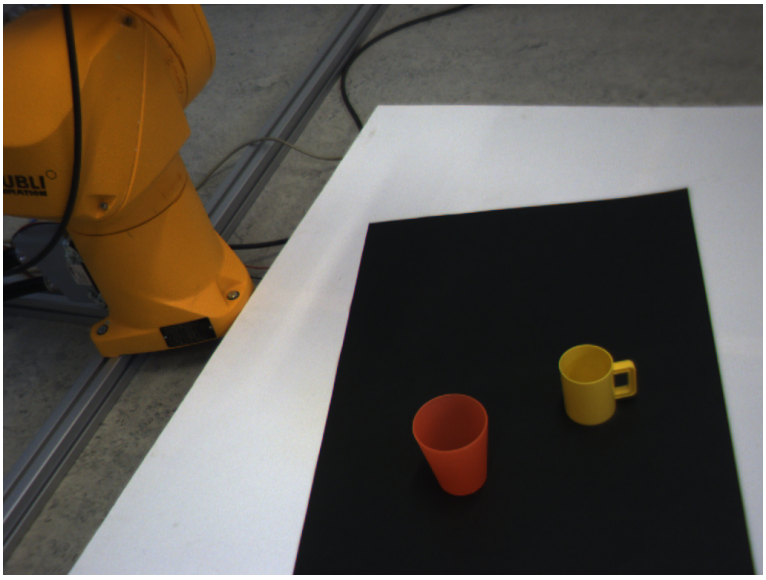


(b)

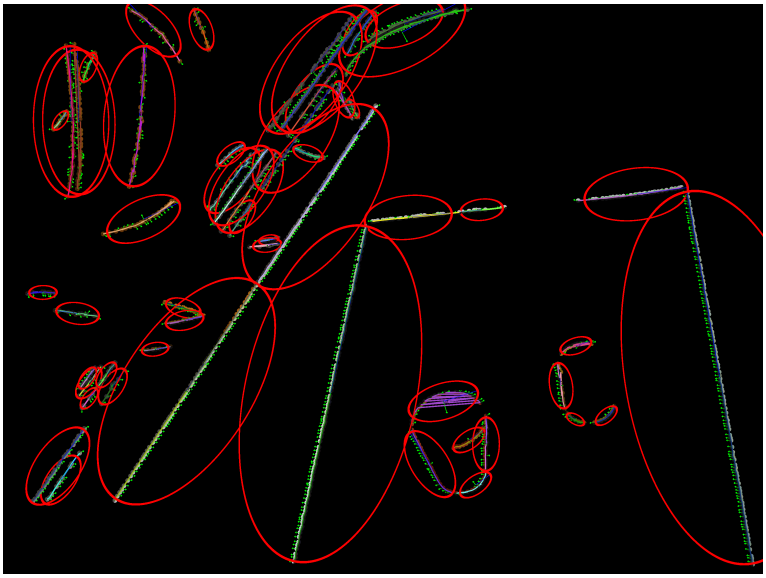


(c)

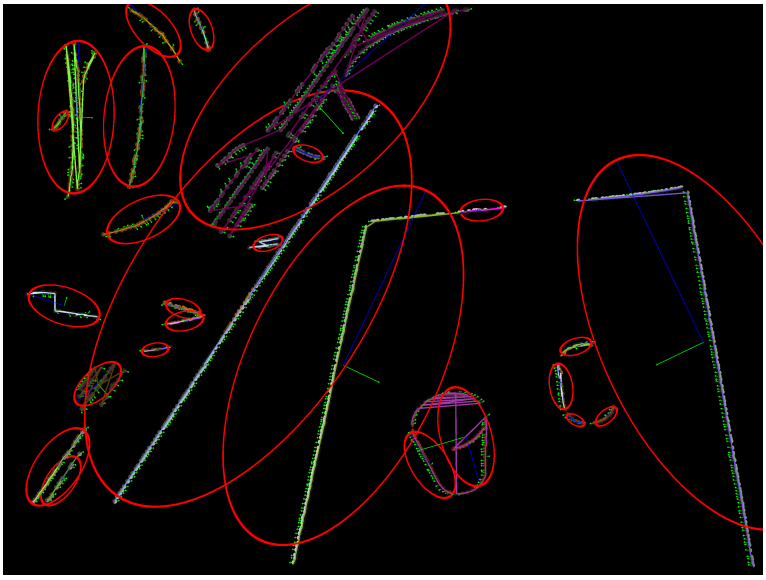
Figure 5: Example of the extraction of Visual Gestalts using good continuation (b) and coplanarity + co–colourity (c) (the red ellipses show the Gestalts).



(a)



(b)



(c)

Figure 6: Example of the extraction of Visual Gestalts using good continuation (b) and coplanarity + co–colourity (c) (the red ellipses show the Gestalts).

These results show that relations, when extended to collinear groups becomes stronger predictor of object structure than when applied to basic primitives. Although the group relations are directly based on the primitives' relations defined in [15], this extension offer a considerably lower likelihood of accidental occurrence.
From these preliminary results, we propose to design a hierarchical architecture for representing explicitly complex structures in the scene and evaluating the saliency thereof.

3D contours extraction: using the process explained in [15], we propose to extract contours from the image representation provided by the primitives.

Evaluation of inter–contour relations in our case we will limit to 1) parallelism + co–colority; and 2) coplanarity + co–colority. Future work should focus on integrating symmetry and the relations provided by the addition of junction primitives (see [16]) into the scheme.

Design good structure to represent shapes As a result of the above-mentioned mechanism, strongly structured objects should appear as densely linked in the resulting graph. If we consider the simple case of a coloured square, we would have each side of the square as a contour. Opposed sides would be parallel and contiguous sides would be coplanar. The advantage of a shape representation based on 3D–contours is that it is largely independent from viewpoint,scaling and sampling. For example, [26] proposed to use a similar hierarchical shape representation for the purpose of object recognition.

Feedback the information to lower level processes E.g. the stereopsis. A stereopsis of good quality is essential for the grouping process to perform well. On the other hand, at each level of the grouping hierarchy new information is obtained that could be used to disambiguate stereopsis, in a similar way that the lowest level grouping information was used in [24]. Chung and Nevatia [2] used a similar approach to stereo disambiguation with the notable difference that they restricted themselves to monocular grouping. We argue here that perceptual grouping and 3D–reconstruction should be processed in parallel using extensive communication between the two processes.
Our objective is to address these points in the upcoming year, in order to obtain a higher level symbolic scene representation.

References

[1] A. Baumberg. Reliable Feature Matching across Widely Separated Views. In *Proc. Conf. Computer Vision and Patter Recognition*, pages 774–781, 2000.

[2] R. Chung and R. Nevatia. Use of monucular groupings and occlusion analysis in a hierarchical stereo system. *Computer Vision and Image Understanding*, 62(3):245–268, 1995.

[3] Cordelia Schmid and Roger Mohr and Christian Baukhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.

[4] D. Crevier. A probabilistic method for extracting chains of collinear segments. *Computer Vision and Image Understanding*, 76(1):36–53, october 1999.

[5] David G. Lowe. Distinctive Image Features from Scale–Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004.

[6] J. H. Elder. Are edges incomplete ? *International Journal of Computer Vision*, 34:97–122, 1999.

[7] J. H. Elder, A. Krupnik, and L. A. Johnstone. Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):661–674, 2003.

[8] J. H. Elder and S. H. Zucker. Evidence for boundary specific grouping. *Vision Research*, 38(1):143–152, 1998.

[9] O. Faugeras. *Three–Dimensional Computer Vision: A Geometric ViewPoint*. MIT Press, 1993.

[10] Frederik Schaffalitzky and Andrew Zisserman. Multi–view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". *Lecture Notes in Computer Science*, 2350:414–431, 2002. in Proceedings of the BMVC02.

[11] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE-PAMI*, 13(9):891–906, 1991.

[12] W. Geisler, J. Perry, B. Super, and D. Gallogly. Edge co–occurrence in natural images predicts contour grouping performance. *Vision Research*, 41:711–724, 2001.

[13] C. G. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[14] J. J. Koenderink and A. J. van Doorn. Representation of Local Geometry in the Visual System. *Biological Cybernetics*, 55:367–375, 1987.

[15] S. Kalkan, N. Pugeault, and N. Krüger. Perceptual operations and relations between 2d or 3d visual entities. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-3, 2007.

[16] S. Kalkan, S. Yan, F. Pilz, and N. Krüger. Improving junction detection by semantic interpretation. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.

[17] P. Kovesi. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3):1–26, 1999.

[18] N. Krüger. Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129, 1998.

[19] N. Krüger and M. Felsberg. A continuous formulation of intrinsic dimension. *Proceedings of the British Machine Vision Conference*, pages 261–270, 2003.

[20] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence & the Simulation of Behavious, AISB Journal*, 1(5):417–427, 2004.

[21] N. Krüger, N. Pugeault, and F. Wörgötter. Multi-modal primitives: Local, condensed, and semantically rich visual descriptors and the formalization of contextual information. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-4, 2007.

[22] Luc Van Gool and Theo Moons and Dorin Ungureanu. Affine / Photometric Invariants for Planar Intensity Patterns. *Lecture Notes In Computer Science*, 1064:642–651, 1996. in Proceedings of the 4th European Conference on Computer Vision — Volume 1.

[23] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[24] N. Pugeault, F. Wörgötter, , and N. Krüger. Structural Visual Events. In *Technical report of the Robotics Group, Maersk Institute, University of Southern Denmark*, number 2007-1, 2007.

[25] N. Pugeault, F. Wörgötter, and N. Krüger. Multi-modal scene reconstruction using perceptual grouping constraints. In *Proc. IEEE Workshop on Perceptual Organization in Computer Vision (in conjunction with CVPR'06)*, 2006.

[26] A. Selinger and R. C. Nelson. A perceptual grouping hierarchy for appearance based 3d object recognition. *Computer Vision and Image Understanding*, 76(1):83–92, october 1999.

Multi–modal Scene Reconstruction using Perceptual Grouping Constraints

5th IEEE Computer Society Workshop on Perceptual Organization in Computer Vision

Nicolas Pugeault
University of Edinburgh
npugeaul@inf.ed.ac.uk

Florentin Wörgötter
University Göttingen
worgott@chaos.gwdg.de

Norbert Krüger
Aalborg University Copenhagen
nk@imi.aau.dk

Abstract

In this work we propose a scheme integrating perceptual grouping into stereopsis to reduce the ambiguity of those early processes. We propose a simple perceptual grouping algorithm that – in addition to the geometric information – makes use of a novel multi–modal affinity measure between local primitives. We then use this group information to 1) disambiguate the stereopsis by enforcing that stereo matches preserve groups; and 2) correct the reconstruction error due to the image pixel sampling using a linear interpolation over the groups. We show quantitative and qualitative demonstrations of those processes on a variety of sequences.

1. Introduction

We propose in this paper an approach using feedback between two mid–level processes, namely perceptual grouping and stereopsis to reduce the ambiguity omnipresent at this level of processing. We base our framework on a novel image representation based on multi–modal local image descriptors called *primitives*, introduced by [21] and applied to stereo by [20]. In this work, we will focus on primitives describing line structures, and we propose a perceptual grouping mechanism which makes use of this rich multi–modal information.

Perceptual grouping can be divided in two tasks: 1) defining an affinity measure between primitives and use it to build a graph of the connectedness between the primitives, and 2) extracting groups, which are the connected components of this graph. We will only define the affinity measure between primitives, and not extract the groups themselves explicitly, as we only need the local grouping information for a primitive to apply the correction mechanisms we propose in this paper. Similar affinity measures have been proposed by [27, 26], which formalised a good continuation constraint, or [9] which included the intensity on each side of the curve into a Bayesian formulation of grouping. Yet in this paper we propose a multi–modal similarity measure, composed of phase, colour and optical flow measurement, and combine it with a classical good continuation criterion forming a novel multi–modal definition of the affinity between primitives. Note that an explicit description of the groups could be extracted easily using a variety of techniques including: normalised [34] or average cuts [32], affinity normalisation [27], dynamic programming [33], *etc*.

The interest of using perceptual organisation in the spatial and temporal domains has been outlined by [31]. Here, we will study how this perceptual grouping information can be used to disambiguate stereopsis and 3D reconstruction using primitives. If we assume that a contour of the image is likely to be a projection of a contour of the 3D scene, then we can expect each 3D contour of the scene to project as a 2D contour on each camera plane (except in the case of occlusion). Conversely, this also implies that any contour in one image has a corresponding contour in the second image (or it is occluded). Thus we will propose an *external* stereo confidence which estimates how well primitives that are part of the same group agree with a putative stereo–match. This allows to discard a large number of potential stereo–correspondences hence reducing the ambiguity of the stereo matching and of the scene reconstruction processes.

We will test this scheme with four different calibrated stereo sequences, illustrated in figure 1. For sequences (a) (b) and (c) we have depth values obtained from a range scanner. Ten different frames from those three sequences were used for quantification in this paper. Sequence (d) was recorded outdoors in a moving car. for which we will show qualitative results.

The novel contributions of this paper are

- a 2D grouping that uses geometric and appearance based information,
- using the 2D grouping for improving stereo matching from a very local level (in contrast to, e.g., [30], where more elaborate features, like ribbons, were considered),

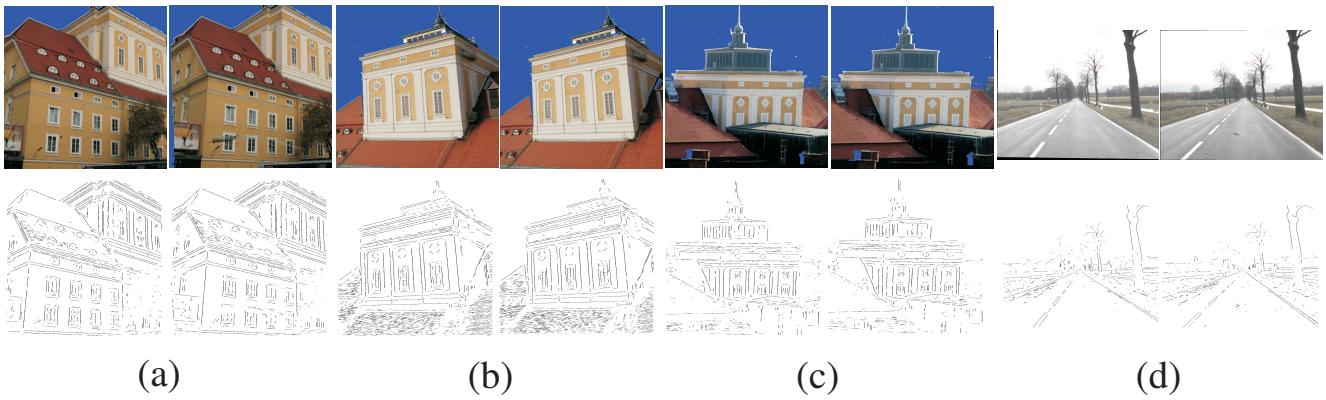


Figure 1. The four sequences on which we tested our approach.

- applying an interpolation method that leads to more reliable estimates of 3D position and 3D-orientation.

The grouping is part of an early cognitive vision framework including ego-motion estimation and temporal accumulation (for an outline see [37]).

The paper is structured as follows: Section 2 will present the image primitives on which we are basing our processing. In section 3, we define the affinity between two primitives. In section 4 we present a stereo–matching process based on primitives similar to [20]. Then in section 5 we propose a simple scheme to 1) increase the reliability of matching and 2) smooth the reconstruction of a stereo sequence using information gained from the perceptual grouping defined earlier.

2. 2D–primitives

Numerous feature detectors exist in the literature (see [22] for a review). Each feature based approach can be divided into an interest point detector (e.g. [3, 4]) and a descriptor describing a local patch of the image at this location, that can be based on histograms (e.g. [6, 22]), spatial frequency [28], local derivatives [15, 13, 1] steerable filters [36], or invariant moments ([23]). In [22] these different descriptors have been compared, showing a best performance for SIFT-like descriptors.

The primitives we will be using in this work are local, multi–modal edge descriptors that were introduced in [21]. In contrast to the above mentioned features these primitives focus on giving a semantically and geometrically meaningful description of the local image patch. The importance of such a semantic grounding of features for a general purpose vision front–end, and the relevance of edge–like structures for this purposes were discussed in [7].

The edge map and the local phase are computed using the monogenic signal (see [11]), although some other kind of filtering could alternatively be used (e.g., steerable filters [36]). The primitives are extracted sparsely at locations in the image that are the most likely to contain edges. This likelihood is computed using the intrinsic dimensionality measure proposed in [19]. The sparseness is assured using a classical winner take all operation, insuring that the generative patches of the primitives do not overlap. Each of the primitive encodes the image information contained by a local image patch of a same size ρ as the kernel used by the filtering operation. Multi–modal information is gathered from this image patch, including the position $\boldsymbol{m}$ of the centre of the patch, the orientation θ of the edge, the phase ω of the signal at this point, the colour $\boldsymbol{c}$ sampled over the image patch on both sides of the edge and the local optical flow $\boldsymbol{f}$, computed using the classical Nagel algorithm (see [25]). Consequently a local image patch is described by the following multi–modal vector:

$$\boldsymbol{\pi} = (\boldsymbol{m}, \theta, \omega, \boldsymbol{c}, \boldsymbol{f}, \rho)^T \tag{1}$$

that we will name *primitive* in the following. The set of primitives describing the stereo images is called *image representation* and written $\boldsymbol{\mathcal{I}}^l$ and $\boldsymbol{\mathcal{I}}^r$ for the images from the left and right camera. The image representation extracted from one image is illustrated in figure 2.

Note that these primitives are of lower dimensionality than, e.g., SIFT (10 vs. 128) and therefore suffer of a lesser distinctiveness. Nonetheless, we will show in section 4 that they are distinctive enough for a reliable stereo matching if the epipolar geometry of the cameras is known. Advantageously, the rich information carried by the 2D–primitives can be reconstructed in 3D, providing a more complete scene representation. Having geometrical meaning for the primitive allows to describe the relation between proximate primitives in terms of perceptual grouping.

3. Perceptual Grouping of 2D–Primitives

Decades ago, the Gestalt psychologists proposed a series of axioms describing the way the human visual system binds together features in an image (see [16, 35, 17]). This process is generally called *perceptual grouping* the Gestalt psychologists proposed that it was driven proper-

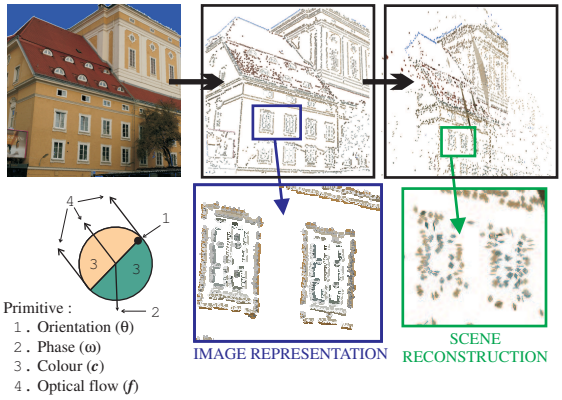


Figure 2. Illustration of the primitive extraction process from a video sequence. The figure shows one image from the sequence (a) from figure 1, on the right, then the 2D–primitives extracted from this image (see section 2), and finally the 3D–primitives reconstructed from the stereo–matches as described in section 4. The bottom row shows a description of the graphic representation of the 2D–primitives, as well as a magnification of the image representation and the reconstructed entities. Note that the structure reconstructed is quite far from the cameras, leading to a certain imprecision in the reconstruction of the 3D–primitives. We will propose a simple scheme addressing this problem in section 5.3

ties like proximity, good continuation, similarity, symmetry, amongst others. More recently, psychophysical experiments measured the impact of different cues for perceptual grouping (see, e.g., [12]). Furthermore, Brunswik and Kamiya [2] proposed that those processes should be related to statistics of natural images, which has been recently confirmed by several studies [18, 8, 14].

We previously defined the primitives as local edge descriptors, and that a group of primitives describe a contour of the image. The Gestalt rule of *proximity* implies that primitives that are closer to one another are most likely to lie on the same contour. According to the Gestalt rule of *good continuation*, we will consider that contours in the image are smooth, and therefore that two proximate primitives in a group will be nearly either collinear or co–circular. In this formulation, a strong inflexion in a contour will lead this contour to be described as *two* groups joining at the inflection point. Furthermore the position and orientation of primitives that are part of a group are the local tangents to the contour described by this group. Finally, the rule of *similarity* states that primitives that are similar (in terms of the colour, phase and optical flow modalities) are most likely to be grouped together. Also, we would expect such properties as colour on both side of a contour to change smoothly along this contour.

The two first cues are joined into a *Geometric constraint* that we describe in section 3.1 and the multi–modal similarity cue is detailed in section 3.2. These two measures are combined into an overall affinity measure that we describe in section 3.3.

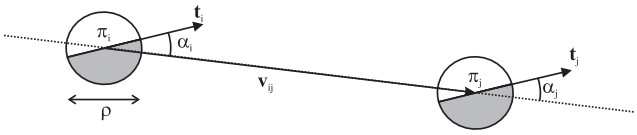


Figure 3. Illustration of the values used for the collinearity computation. If we consider two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$, then the vector between the centres of these two primitives is written $\boldsymbol{v}_{ij}$, and the orientations of the two primitives are designated by the vectors t_i and t_j, respectively. The angle formed by $\boldsymbol{v}_{ij}$ and t_i is written α_i, and between $\boldsymbol{v}_{ij}$ and t_j is written α_j. ρ is the radius of the image patch used to generate the primitive.

3.1. Geometric constraint

If we consider two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$ in $\mathcal{I}$, then the likelihood that they both describe the same contour can be formulated as a combination of three basic constraints on their relative position and orientation — see figure 3.

Proximity (c_p []):

$$c_p\left[g_{i,j}\right] = 1 - e^{-\max\left(1-\frac{||\mathbf{v_{i,j}}||}{\rho\tau},0\right)} \tag{2}$$

Here, ρ stands for the radius of the the primitives in pixels. $\rho\tau$ is the size of the neighbourhood considered in pixels. $\|\mathbf{v_{i,j}}\|$ is the distance in pixels separating the centres of the two primitives.

Collinearity (c_{co} []):

$$c_{co}\left[g_{i,j}\right] = 1 - \left|\sin\left(\frac{|\alpha_i| + |\alpha_j|}{2}\right)\right| \tag{3}$$

Here α_i and α_j are the angles between the line joining the two primitives centres and the orientation of, respectively, $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$.

Co–circularity (c_{ci} []):

$$c_{ci}\left[g_{i,j}\right] = 1 - \left|\sin\left(\frac{\alpha_i + \alpha_j}{2}\right)\right| \tag{4}$$

The combination of those three criteria forms the *geometric* affinity measure:

$$\mathbf{G}_{i,j} = \sqrt[3]{c_e\left[g_{i,j}\right] \cdot c_{co}\left[g_{i,j}\right] \cdot c_{ci}\left[g_{i,j}\right]} \tag{5}$$

where $\mathbf{G}_{i,j}$ is the geometric affinity between two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$. This affinity represent the likelihood for a curve having for tangents those two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_i$ to be an actual contour of the scene.

3.2. Multi–modal Constraint

Effectively, the more similar are the modalities between two primitives, the more likely are those two primitives to lie on the same contour. Note that [8] already proposed to use the intensity as a cue for perceptual grouping, yet here

we use a combination of the phase, colour and optical flow modalities of the primitives to decide if they describe the same contour:

$$\mathbf{M}_{i,j} = 1 - w_\omega d_\omega\left(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j\right) - w_{\boldsymbol{c}} d_{\boldsymbol{c}}\left(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j\right) - w_{\boldsymbol{f}} d_{\boldsymbol{f}}\left(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j\right) \tag{6}$$

where d_ω is the phase distance, $c_{\boldsymbol{c}}$ the colour distance and $c_{\boldsymbol{f}}$ the optical flow distance between the two primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$. These metrics are similar to the ones used in [29, 20]. w_ω, $w_{\boldsymbol{c}}$ and $w_{\boldsymbol{f}}$ are the relative weight of the modalities, such that $w_\omega + w_{\boldsymbol{c}} + w_{\boldsymbol{f}} = 1$.

3.3. Primitive Affinity

The overall affinity between all primitives in an image is formalised as a matrix $\boldsymbol{A}$, where $\boldsymbol{A}_{i,j}$ holds the affinity between the primitives $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$. We define this affinity from equations (5) and (6), such that 1) two primitives complying poorly with the good continuation rule have an affinity close to zero; and 2) two primitives complying with the good continuation rule yet strongly dissimilar will have only an average affinity. The affinity is formalised as follows:

$$c\left[g_{i,j}\right] = \boldsymbol{A}_{i,j} = \sqrt{\mathbf{G}\left(\alpha \mathbf{G}_{i,j} + (1-\alpha)\mathbf{M}_{i,j}\right)} \tag{7}$$

where α is the weighting of geometric and multi–modal (*i.e.* phase, colour and optical flow) information in the affinity. A setting of $\alpha = 1$ implies that only geometric information (proximity, collinearity and co-circularity) is used, while $\alpha = 0$ indicates that geometric and multi–modal information are evenly mixed. The groups generated for the left and right frames for each sequence are drawn in figure 1, bottom row. Dark lines describe strings of grouped primitives. One can see in those images that the major contours of the images are adequately described.

4. Stereopsis using 2D–primitives

Classical stereopsis allows reconstructing a 3D point from two corresponding stereo points. A review of stereo–algorithms was presented in [24], dense two frames stereo algorithms were also compared in [5]. In these papers the different algorithms were compared on mainly artificial images, with a disparity d that ranges in $0 \leq d \leq 16$. In this work we make use of a sparse, feature based representation, applied on high resolution video sequences of natural scenes, where the ground truth was obtained using a range scanner. The allowed disparity range for these scenes is $0 \leq d \leq 200$, leading to a comparable level of ambiguity (*i.e.* between 10 and 20 candidates depending on the primitive being matched).

The stereopsis used for this paper is a simple local winner–take–all scheme: all primitives in the right image that lie on the epipolar line are *potential correspondences* and their individual likelihood is set as their multi–modal similarity with the original primitive in the left image. Then the most similar primitive is taken as the most likely correspondence. The multi–modal distance between two primitives is defined as a linear combination of the modal distances between the two primitives:

$$d_m(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j) = \sum_m w_m d_m(\boldsymbol{\pi}_i, \boldsymbol{\pi}_j) \tag{8}$$

where w_m is the relative weighting of the modality m, with $\sum_m w_m = 1$ (we use distance functions for the modalities that are similar to the ones proposed in [29, 20]).

In figure 6(a) the ROC curves showing the performance of the stereo–matching when using as likelihood estimation the similarities in each of the modalities held by a primitive, alongside with the performance of the multi–modal distance proposed in equation (8). We can see that: 1) all modalities offer a discrimination better than chance between correct and erroneous correspondences; and 2) the multi–modal distance offers a better discrimination than the individual modalities. In this figure we can see that the colour modality is a particularly strong discriminant for stereopsis. This is explained by the fact that the hue and saturation are sampled on each side of the edge, leading to a 4–dimensional modality, where phase and orientation are only 1–dimensional and optical flow is 2–dimensional (albeit the aperture problem reduces it to one effective dimension: the normal flow). On the other hand the poor performance of the optic flow modality could be explained by the relative simplicity of the motion in this scene: a pure forward translation of the camera, with no moving object. Therefore, we would expect the performance of individual modalities to vary depending on the scenario, and the robustness of the multi–modal constraint could be further enhanced by a contextual weighting. Nevertheless, in a variety of scenarios the use of a static weighting proved robust enough to obtain reliable stereopsis.

Moreover, by making use of the rich semantic information carried by the primitives, the stereopsis yield a set of geometrically meaningful entities rather than an mere disparity map We call the reconstructed entities 3D–primitives $\boldsymbol{\Pi}$:

$$\boldsymbol{\Pi} = (\boldsymbol{M}, \boldsymbol{\Theta}, \Omega, \boldsymbol{C})^T \tag{9}$$

where $\boldsymbol{M}$ is the location in space, $\boldsymbol{\Theta}$ is the 3D orientation of the edge, Ω is the phase across this edge, and $\boldsymbol{C}$ holds the colour information for this edge — see attached material. In figure 7(a) we show the 3D–primitives that were reconstructed after a stereo–matching based on the multi–modal confidence from equation (8).

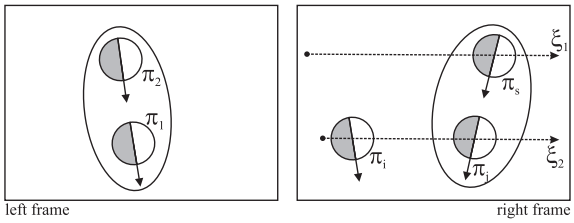


Figure 4. The BSCE criterion: Let $\boldsymbol{\pi}_1$ be a primitive in the left frame forming a group with a second primitive $\boldsymbol{\pi}_2$. $\boldsymbol{\pi}_2$ has a stereo correspondence $\boldsymbol{\pi}_s$ in the right image. Both $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$ in the right image lie on the epipolar line ξ_1 of $\boldsymbol{\pi}_1$; hence these two primitives are both putative correspondences of $\boldsymbol{\pi}_1$. Furthermore, the primitive $\boldsymbol{\pi}_i$ is clearly the most similar to $\boldsymbol{\pi}_1$ (due to a closer orientation), hence this stereo–correspondence $s_{1\to i}$ yield a higher multi–modal confidence than would, e.g. $s_{1\to j}$. Yet, when considering the BSCE criterion we realise that only the putative correspondence $\boldsymbol{\pi}_j$ forms a group $g_{j,s}$ with $\boldsymbol{\pi}_s$, conserving the group relation $g_{1,2}$ between $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$.

5. Perceptual Grouping Constraints to Improve Stereopsis

In addition to their richness, primitives are very redundant along contours, and this redundancy allows us to use perceptual grouping to derive the following two constraints for the matching process:

Isolated primitives are likely to be unreliable: As primitives are extracted redundantly along the contours, conversely an isolated primitive is likely to be an artifact. Hence isolated primitives can be neglected.

Stereo consistency over groups: If a set of primitives forms a contour in the first image, the *correct correspondences* of these primitives in the second image also form a contour.

5.1. Basic Stereo Consistency Event (BSCE)

As explained in section 3, 2D–primitives represent local estimators of image contours. A constellation of those 2D–primitives describe the contour as a whole. Those contours are consistent over stereo, with the notable exception of partially occluded contours — see figure 1, bottom row. Hence, if two primitives describe a contour in one image then their correspondences in the second image should also describe the same contour, and those two 2D contours are the projection of the same 3D contour onto the two different optical planes. In section 3, we defined the likelihood for two primitives to describe the same contour as the affinity between these two primitives, hence we can rewrite the previous statement as:

Given two primitives $\boldsymbol{\pi}_i^l$ and $\boldsymbol{\pi}_j^l$ in $\boldsymbol{\mathcal{I}}^l$ and their respective correspondences $\boldsymbol{\pi}_n^r$ and $\boldsymbol{\pi}_p^r$ in a second image $\boldsymbol{\mathcal{I}}^r$; if $\boldsymbol{\pi}_i^l$ and $\boldsymbol{\pi}_j^l$ belongs to the same group in $\boldsymbol{\mathcal{I}}^l$ then $\boldsymbol{\pi}_n^r$ and $\boldsymbol{\pi}_p^r$ should also be part of a group in $\boldsymbol{\mathcal{I}}^r$. — see figure 4.

We call the conservation of the link between a pair of primitives in the stereo–correspondences of those primitives the *Basic Stereo Consistency Event* (BSCE).

This condition can then be used to test the validity of a stereo–hypothesis. Consider a primitive $\boldsymbol{\pi}_i^l$, and a stereo hypothesis:

$$s_{i\to n} : \boldsymbol{\pi}_i^l \to \boldsymbol{\pi}_n^r \tag{10}$$

and consider a neighbour $\boldsymbol{\pi}_j^l \in N(\boldsymbol{\pi}_i^l)$ of $\boldsymbol{\pi}_i^l$ such that the two primitives share an affinity $c\,[g_{i,j}]$. For this second primitive a stereo–correspondence $\boldsymbol{\pi}_p^n$ with a confidence of $c\,[s_{j\to p}]$ exists. We can then estimate how well the stereo–hypothesis $s_{i\to n}$ preserves the BSCE:

$$E(g_{i,j}, s_{i\to n}) = \begin{cases} \sqrt{c\,[s_{j\to p}] \cdot c\,[g_{i,j}]} & if\ c\,[g_{n,p}] > \varepsilon \\ -\sqrt{c\,[s_{j\to p}] \cdot c\,[g_{i,j}]} & else \end{cases} \tag{11}$$

In other words, considering a stereo–pair of primitives: the BSCE of a primitive in the first image with one of its neighbour is high if they share a strong affinity and if this second primitive creates a stereo–hypothesis such that the correspondences in the second image of both primitives *also* share a strong affinity. It is low if the stereo–correspondence of this primitive and the stereo–correspondences of other primitives part of the same group, do not form a group in the other image. This naturally extends the concept of group as defined in section 3 into the stereo domain.

5.2. Neighbourhood Consistency Confidence

Building on the formula (11), we can define how *the whole neighbourhood* of a primitive is consistent with a given stereo hypothesis.

The previous formula tells us how a 2D–primitive stereo correspondence is consistent with our knowledge of the set of stereo hypotheses for a second 2D–primitive, in its neighbourhood. Now, if we consider a primitive $\boldsymbol{\pi}_i^l$ and an associated stereo–correspondence $s_{i\to n}$, we can integrate this BSCE confidence over the neighbourhood of the primitive $\boldsymbol{\mathcal{N}}_i^l$ — as defined in section 3.3.

$$c_{ext}[s_{i\to n}] = \frac{1}{\#\boldsymbol{\mathcal{N}}_i^l} \sum_{\boldsymbol{\pi}_k^l \in \boldsymbol{\mathcal{N}}_i^l} E(\boldsymbol{\pi}_1^l, \boldsymbol{\pi}_k^l, s_{i\to n}) \tag{12}$$

Where $\#\boldsymbol{\mathcal{N}}_i^l$ is the size of the neighbourhood — *i.e.* the number of neighbours of $\boldsymbol{\pi}_1^l$ considered. We call this new confidence the *external confidence* in $s_{i\to n}$, as opposed to the internal confidence given by the multi–modal similarity between the 2D–primitives— equation (8). In figure 5, one can see that the correct correspondences have mostly positive external confidences, while incorrect ones have mainly negative values. Therefore, applying a threshold on the external confidence will remove stereo hypotheses that are inconsistent with their neighbourhood, and thus reduce the ambiguity of the stereo–matching. Note that selecting a

threshold higher than zero implies the removal of all the isolated primitives (as an isolated primitive has an external confidence of zero by definition).

Figure 6(b) shows ROC curves of the performance for varying thresholds on the multi–modal similarity. Each of the curve drawn shows the performance for different thresholds (respectively threshold values of $-0.6, -0.3, 0, +0.3$, and without threshold) applied to the external confidence prior to the ROC analysis. We can see from those results that applying a bias on the decision based on the external confidence is improving significantly the accuracy of the decision process. Depending on the type of selection process desired — very selective and reliable, or more lax, but yielding a denser set of correspondences — another threshold can be chosen. The best overall improvement seems to be reached for a threshold of -0.3 over the external confidence. Nonetheless, when we consider a case where very high reliability is required, a threshold of 0 (meaning discarding all primitives which are part of no group) might be preferred. Note that when a threshold is applied to the external confidence prior to the ROC analysis, the resulting curve do not reach the $(1, 1)$ point of the graph. This is normal as the threshold already remove some stereo–hypotheses even before the multi–modal confidence is considered.

The 3D–primitives reconstructed after such a scheme are shown in figure 7(b).

5.3. Interpolation in Space

One issue when reconstructing 3D structures from stereopsis is that the accuracy of the reconstructed entities is decreasing with the distance to the cameras, due to the pixel sampling of the images — see [10]. Figure 7(b) shows the reconstruction of the tree (along with the road markings) in sequence (d) — see figure 1. There we can see that, although all primitives describe the contour of the tree from the same point of view, their exact position and orientation in space vary, and they certainly do not form a contour in space.

Yet, we do know that the 2D–primitives they are reconstructed from a group in both stereo images (*c.f.* section 5 and figure 1 bottom row), and as such that they form a smooth continuous contour. Hence we can assume that they are the projection on the image planes of a smooth and continuous contour of the scene (except in some extreme cases and under rare viewpoints), and as such that the reconstructed 3D–primitives should also describe such a curve.

A common way of reducing such noise in the sampling of a smooth function is to use linear smoothing, hence we propose to apply it to the 3D–primitives. For each iteration n of this smoothing, the position $\boldsymbol{M}$ and orientation $\boldsymbol{\Theta}$ of the primitive $\boldsymbol{\Pi}_i^{(n)}$ are changed to the average between their previous values $\boldsymbol{\Pi}_i^{(n-1)}$ and values interpolated from the primitives reconstructed out of the two closest neighbours of the 2D–primitive in the images $I(\boldsymbol{\Pi}_j^{(n-1)}, \boldsymbol{\Pi}_k^{(n-1)})$.

$$\boldsymbol{M}_i^{(n)} = \frac{1}{2}\left(\boldsymbol{M}_i^{(n-1)} + I(\boldsymbol{M}_j^{(n-1)}, \boldsymbol{M}_k^{(n-1)})\right) \quad (13)$$

$$\boldsymbol{\Theta}_i^{(n)} = \frac{1}{2}\left(\boldsymbol{\Theta}_i^{(n-1)} + I(\boldsymbol{\Theta}_j^{(n-1)}, \boldsymbol{\Theta}_k^{(n-1)})\right) \quad (14)$$

Figure 7 illustrate the reconstructed 3D–primitives from the sequence (d) (*c.f.* figure 1). Note that it is necessary to choose a point of view sufficiently different from the one of the camera to highlight the reconstruction errors, while being sufficiently similar for the shapes of the scene to be recognisable. We chose a point of view located high on the right side of the scene, looking downwards at the road.

When comparing figures 7(a) and 7(b) we can see that a large number of outliers are discarded from the reconstructed 3D–primitives, leading to a cleaner description of the scene. Figure 7(c) shows the same part of the scene (d) after 3 iterations of the linear smoothing. The 3D–primitives forming the contour of the tree and the road markings are now smoothly aligned.

6. Conclusion

In this paper we defined an affinity relation between image primitives making use of the rich multi–modal information available. Therefore the resulting affinity measure encompass more than just the good continuation cue but also continuity in phase, colour and optical flow. We have illustrated that, on varied sequence, the resulting groups follow adequately the contours of the image. In a second part we proposed a simple measure of the conservation of those groups, and hence of the neighbourhood structure of a primitive, across stereo. Using this conservation we could formalise a contextual estimation of the likelihood of a stereo correspondence. We show that using this new external confidence measure in conjunction with a similarity measure we can improve significantly the performance of the stereo–matching process. Furthermore, we show that interpolation can be used over a group to correct the smoothness of the reconstructed representation.

Acknowledgement: We thank the company Riegl for the images with known ground truth used for sequence (a), (b) and (c). This work described in this paper was part of the European project ECOVISION.

References

[1] A. Baumberg. Reliable Feature Matching across Widely Separated Views. In *Proc. Conf. Computer Vision and Patter Recognition*, pages 774–781, 2000. 2

[2] E. Brunswick and J. Kamiya. Ecological cue validity of 'proximity' and other gestalt factors. *Journal of Psychology*, 66:20–32, 1953. 3

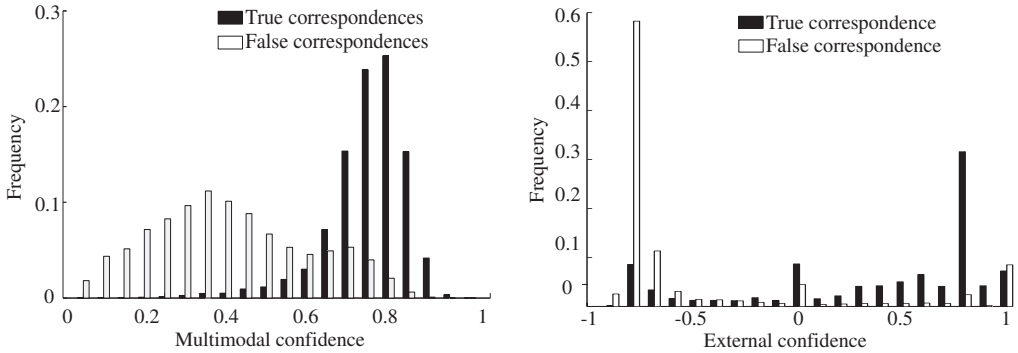
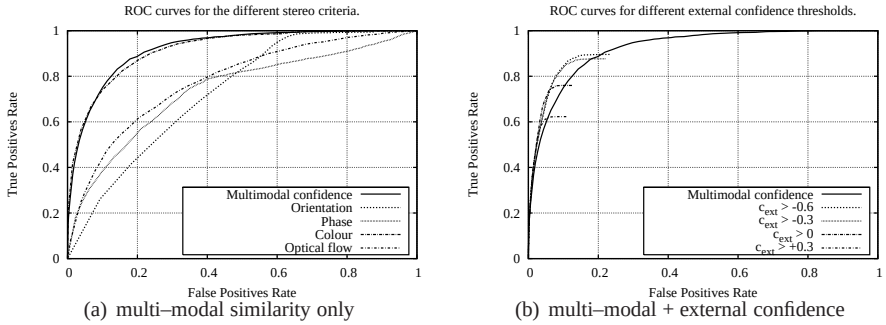


Figure 5. Distribution of multi–modal similarity and external confidence for correct (black bars) and false (white bars) correspondences. These data have been collected over 10 frames of the sequences (a), (b) and (c) — see figure 1.



(a) multi–modal similarity only (b) multi–modal + external confidence

Figure 6. ROC curves for the performance of the multi–modal confidence to discriminate correct from erroneous correspondences. (a) Comparisons of the different modalities for stereo–matching (see for a discussion of the role of colour in the text). (b) Each curve stands for a the application of a different threshold over the external confidence, prior to the ROC analysis. Those curves represent the statistics over 10 frames of the two sequences with ground truth — see figure 1.

[3] Chris Harris and Mike Stephens. A Combined Corner and Edge Detector. In *Proceedings of Alvey Conference*, pages 189–192, 1987. 2

[4] Cordelia Schmid and Roger Mohr and Christian Baukhage. Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000. 2

[5] Daniel Scharstein and Richard Szeliski. A Taxonomy and Evaluation of Dense Two–Frame Stereo Correspondence Algorithms. *International Journal of Computer Vision*, 47(1/2/3):7–42, 2002. 4

[6] David G. Lowe. Distinctive Image Features from Scale–Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004. 2

[7] J. H. Elder. Are edges incomplete ? *International Journal of Computer Vision*, 34:97–122, 1999. 2

[8] J. H. Elder and R. M. Goldberg. Inferential reliability of contour grouping cues in natural images. *Perception*, 27(11), 1998. 3

[9] J. H. Elder and R. M. Goldberg. Ecological statistics of gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2:324–353, 2002. 1

[10] O. Faugeras. *Three–Dimensional Computer Vision: A Geometric ViewPoint*. MIT Press, 1993. 6

[11] M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 41(12), 2001. 2

[12] D. J. Field, A. Hayes, and R. F. Hess. Contour integration by the human visual system: evidence for a local "association field". *Vision Research*, 33(2):173–193, 1993. 3

[13] Frederik Schaffalitzky and Andrew Zisserman. Multi–view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". *Lecture Notes in Computer Science*, 2350:414–431, 2002. in Proceedings of the BMVC02. 2

[14] W. Geisler, J. Perry, B. Super, and D. Gallogly. Edge co–occurrence in natural images predicts contour grouping performance. *Vision Research*, 41:711–724, 2001. 3

[15] J. J. Koenderink and A. J. van Doorn. Representation of Local Geometry in the Visual System. *Biological Cybernetics*, 55:367–375, 1987. 2

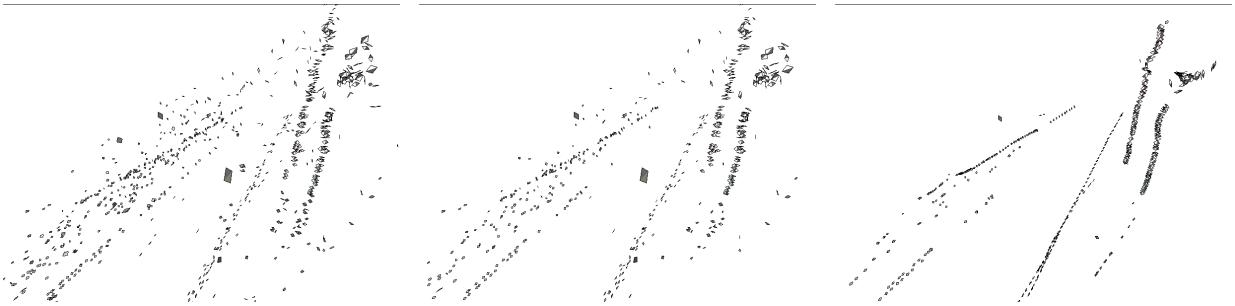
[16] K. Koffka. *Principles of Gestalt Psychology*. Lund Humphries, London, 1935. 2

[17] K. Köhler. *Gestalt Psychology: An introduction to new concepts in psychology*. New York: Liveright, 1947. 2

[18] N. Krüger. Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129, 1998. 3

[19] N. Krüger and M. Felsberg. A continuous formulation of intrinsic dimension. In *Proceedings of the British Machine Vision Conference*, 2003. 2

[20] N. Krüger and M. Felsberg. An explicit and compact coding of geometric and structural information applied to stereo matching. *Pattern Recognition Letters*, 25(8), 2004. 1, 2, 4



(a) Stereo reconstruction (b) After external confidence threshold (c) After linear smoothing

Figure 7. Reconstruction of 3D–primitives from stereo–matches obtained from sequence (d) (*c.f.* figure 1). (a) shows the reconstruction resulting from a stereo–matching done using only the multi–modal stereo approach (with a threshold of 0.4 on the multi–modal confidence). (b) shows reconstruction obtained when an additional threshold of 0 is applied to the external confidence. (c) shows the corrected entities, after 3 iterations of the linear smoothing process.

[21] N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *Interdisciplinary Journal of Artificial Intelligence & the Simulation of Behavious, AISB Journal*, 1(5):417–427, 2004. 1, 2

[22] Krystian Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, Oct. 2005. 2

[23] Luc Van Gool and Theo Moons and Dorin Ungureanu. Affine / Photometric Invariants for Planar Intensity Patterns. *Lecture Notes In Computer Science*, 1064:642–651, 1996. in Proceedings of the 4th European Conference on Computer Vision — Volume 1. 2

[24] Myron Z. Brown and Darius Burschka and Gregory D. Hager. Advances in Computational Stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):993–1008, Aug. 2003. 4

[25] H.-H. Nagel. On the estimation of optic flow: Relations between different approaches and some new results. *Artificial Intelligence*, 33:299–324, 1987. 2

[26] P. Parent and S. W. Zucker. Trace interface, curvature consistency, and curve detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8):823–839, 1989. 1

[27] P. Perona and W. Freeman. A factorization approach to grouping. In *Proceedings of the ECCV*, volume 1406, 1998. 1

[28] Peter Kovesi. Image Features from Phase Congruency. *Videre: Journal of Computer Vision Research*, 1(3), 1999. 2

[29] N. Pugeault and N. Krüger. Multi–modal matching applied to stereo. In *Proceedings of the British Machine Vision Conference 2003*, 2003. 4

[30] Ronald Chung and Ramakant Nevatia. Use of Monocular Groupings and Occlusion Analysis in a Hierarchical Stereo System. *Computer Vision and Image Understanding*, 62(3):245–268, Nov. 1995. 1

[31] S. Sarkar and K. L. Boyer. *Computing Perceptual Organization in Computer Vision*. World Scientific Publishing Co. Pte. Ltd., 1994. 1

[32] S. Sarkar and P. Soundararajan. Supervised learning of large perceptual organization: Graph spectral partitioning and learning automata. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(5):504–525, 2000. 1

[33] A. Sha'ashua and S. Ullman. Grouping contours by iterated pairing network. In *Neural Information Processing Systems (NIPS)*, volume 3, 1990. 1

[34] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000. 1

[35] M. Wertheimer, editor. *Laws of Organsation in Perceptual Forms*. Harcourt & Brace & Javanowitch, London, 1935. 2

[36] William T. Freeman and Edward H. Adelson. The Design and Use of Steerable Filters. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 13(9):891–906, Sept. 1991. 2

[37] F. Wörgötter, N. Krüger, N. Pugeault, D. Calow, M. Lappe, K. Pauwels, M. V. Hulle, S. Tan, and A. Johnston. Early cognitive vision: Using gestalt-laws for task-dependent, active image-processing. *Natural Computing*, 3(3):293–321, 2004. 2