Project no.:	027657
Project full title:	Perception, Action & Cognition through Learning of Object-Action Complexes
Project Acronym:	PACO-PLUS
Deliverable no.:	D9.2.5

Title of the deliverable: Publication of algorithms for learning OACs as open source code

Contractual Date of Delivery to the CEC:	31. May 2010
Actual Date of Delivery to the CEC	05. July 2010
Organisation name of lead contractor for this deliverable:	KIT
Authors:	All
Participants:	All
Work package contributing to the deliverable	WP9, WP1
Nature:	R & D
Version	1.0
Total number of pages:	6
Start date of project:	1st Feb. 2006
Duration:	52 months

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:

This is a new deliverable which was added in the new implementation plan which was submitted to the Commission in January 2010 to justify the project extension.

The deliverable describes the major software components which have been developed in the project for different tasks associated with learning of OACs in PACO-PLUS. The developed software will be made available as **open source** to ensure that the concept of object-action complexes, which has proven to be successful at realizing cognitive abilities on robotic systems, is accessible to a wider scientific and industrial community.

This deliverable consists of a list of the software packages, a brief description of the functionality as well as the links to the software repositories on the WWW. Direct links to all software packages on the PACO-PLUS webpage will be made.

Keyword list: Open source software developed in PACO-PLUS;

Executive Summary

In the course of the project many algorithms have been developed that realize the learning and interplay of object-action complexes. To ensure that the concept of object-action complexes, which has proven to be successful at realizing cognitive abilities on robotic systems, is accessible to a wider scientific and industrial community, the partners will disseminate their algorithms as open-source software. Examples of software that will be made available as open-source code are shown in the Table 1 below.

A documentation of the developed software components is also available to make it easier to be used by researchers and programmers who were not involved in the software development.

All software packages will be linked to the PACO-PLUS webpage.

Table 1: Open-Source Software

Software	**CoVis**
Description	CoViS (Cognitive Vision Software) is a C++ computer vision library framework. Its core functionality is the extraction of an Early Cognitive Vision inspired scene representation. Based on this additional processes (e.g., motion estimation, model accumulation, creation of higher level features) are provided. CoViS is supposed to be used as an Early CognitiveVvision level which bridges the gap between early vision pixel wise processing and cognitive vision tasks and which is in particular useful for bootstrapping processes in cognitive robots. It is the basis for all applications developed by SDU within the PACO-PLUS project. CoViS is made available under a BSD license. Access is currently by personal request since we are still improving the documentation and demo programs. We expect a full public release by fall 2010.
Source code	C++
Partner	SDU
Available	www.covig.org

Software	**BADGr, the BoxGraspingSuite**
Description	BADGr, the BoxGraspingSuite, is a package for Box Approximation, Decomposition, and Grasping. BADGr provides modules to approximate the shape of a point cloud (possibly from sensor data) by box primitives. These box primitives then serve as a base for the generation of box-based pre-grasp hypotheses for robot grippers. BADGr allows to implement algorithms and experiments on shape approximation and grasping, using the Columbia grasp simulator GraspIt!. BADGr uses a client/server interface to the grasp simulator GraspIt! and is therefore not standalone. The program envelops all the experimental setups used for KTH

	publications on BADGr in the PACO-PLUS project.
Source code	C++
Partner	KTH
Available	http://www.csc.kth.se/~khubner/badgr

Software	**Hand Pose Estimator**
Description	Provides real-time estimation of the hand pose in terms of orientation with respect to the camera and joint angles. It is based on classification of a monocular view against a database of hand pose synthetic images generated with the Poser software. After that classification, the best candidates are compared with previous estimations to avoid glitches. The library also shows which is the closest match from the database. More information about the system can be found in: Javier Romero, Hedvig Kjellstrom, Danica Kragic. *Hands in Action: Real-Time 3D Reconstruction of Hands in Interaction with Objects*. In ICRA 2010.
Source code	C++
Partner	KTH
Available	http://www.csc.kth.se/~jrgn/releasePACO.tar.bz2

Software	**Integrating Vision Toolkit (IVT)**
Description	The IVT is an object-oriented computer vision library that offers various functionality that is necessary for developing robot vision systems in a convenient way, such as a camera interface and various ready-to-use image capture modules, a generic camera model, camera calibration, mathematic data structures and functions, various image processing functions for filtering, segmentation, 3D reconstruction, recognition, tracking, and particle filtering. The IVT is a platform independent standalone library with no dependencies to other libraries. It offers its own multi-platform GUI toolkit with implementations for various target platforms (Win32 API, Cocoa for Mac OS X, GTK for Linux, or Qt). All work on computer vision by KIT within PACO-PLUS has been implemented using the IVT. The number of users worldwide steadily increases, and there is a vital forum on the Sourceforge site. The IVT is available under a BSD license.
Source code	C++
Partner	KIT
Available	Sourceforge, http://ivt.sourceforge.net

Software	**SPOAC: Symbolic Planning Interface for Object Action Complexes**
Description	SPOAC provides a network interface and a software framework for symbolic execution of OACs on a robot. It contains network interfaces to a symbolic planner, an OAC database server and a GUI through which all three parts can be controlled and monitored. The robot side framework consists of a modular set of perception handlers and OAC executors. Perception handlers continously update short term memory to have the latest information received from the robot's sensors. This information is translated into a symbolic domain suitable for communication with the planning software. OAC executors make use of the information in short term memory when they require sensory feedback during the execution of an OAC. The SPOAC libraries and services have been implemented in C++ but the network interface built with ICE (Internet Communications Engine) is language agnostic and allows for high level AI software written in any language to be connected to the robot. Such software can then receive symbolic representations of the robot's state and its perception, send symbolic instructions to the robot and interact with the OAC database server.
Source code	C++
Partner	KIT
Available	http://spoac.sourceforge.net

Software	**PLComm: Plan communication library**
Description	This library facilitates the exchange of messages between the high-level planning components (e.g., action planner and plan execution monitor) and other modules in a multi-component software system. The library is divided into three main parts: (i) a lightweight network transport mechanism, based on sockets, providing a range of communication services (i.e., point-to-point, subscription, and broadcast messaging), (ii) a predefined message exchange protocol for requesting and delivering high-level plans to particular components in the system, and (iii) interface code for operating with the PKS planner, and for using plans generated by other external services.
Source code	C++
Partner	UEDIN
Available	http://homepages.inf.ed.ac.uk/rpetrick/

Software	**PEMPKS: Plan execution monitor for PKS**
Description	The PKS plan execution monitor is designed to control plan generation and replanning activities in an online system using the PKS planner. The plan execution monitor makes use of the message passing protocol provided by the

	PLComm library, and includes interface code for the network services provided by PLComm. It also includes sample interface code that can be adapted for other network transport mechanisms (e.g., ICE). The plan execution monitor provides a network-based front-end to the PKS planner and its services (e.g., domain specification, plan generation, and plan exploration), but can also be used in an "offline" mode, to simulate planning and replanning scenarios.
Source code	C++
Partner	UEDIN
Available	http://homepages.inf.ed.ac.uk/rpetrick/

Software	**Planning domain descriptions**
Description	The high-level domain definitions we have developed during PACO-PLUS will be made available as part of a collection of benchmark planning problems. This collection will include descriptions of the planning domains for the SDU robot/vision system and the KIT ARMAR robot. It will also include examples of simple dialog problems. The domain description files will be provided primarily as PKS input files, but also converted to PDDL where possible.
Source code	PKS and PDDL input file formats
Partner	UEDIN
Available	http://homepages.inf.ed.ac.uk/rpetrick/

Software	**Dynamic Motor Primitives for discrete and periodic movements**
Description	Motor knowledge is an integral part of the OAC formalism. It is often represented by dynamic movement primitives (DMPs). The code implementing periodic and discrete dynamic movement primitives as well as the generalizing the DMPs across the space of the observed movements is provided as a Matlab toolbox. The main application is programming of complex robot skills by demonstration, both for industrial and service robots. The software can also be used for educational purposes.
Source code	Matlab
Partner	JSI
Available	http://www.ijs.si/~aude/DMPtoolbox

Software	**IPSA - Inventor Physics Simulation API**
Description	Inventor Physics API (IPSA) is an extension of the Open Inventor toolkit with physics objects using the ODE library for simulating rigid body dynamics. IPSA

	allows the generation of 3D animations and simulations by considering physical attributes such as shape, mass, gravitation, friction etc. The simulations may be programmed directly using the Open Inventor extension classes or more preferably by writing an Open Inventor Scene description file. Several examples for both methods are provided with the code.
Source code	C++
Partner	KIT
Available	http://sourceforge.net/projects/ipsa (ISC License)

Software	**Software for the control of the Karlsruhe Humanoid Head**
Description	Software components needed for the control of the Karlsruhe Humanoid head. This includes an API for the access to the actuators and sensors of the head. This API allows using the head together with applications of other researchers in different labs. The applications range from visual tasks (grasping, object recognition and localization) to human-robot interaction.
Source code	C++
Partner	KIT
Available	PACO-PLUS webpage: www.paco-plus.org and KIT Head Active Humanoid Head Wiki http://head.itec.uka.de/wiki/Welcome