



RESEARCH CENTER

FIELD

Perception, Cognition and Interaction

Activity Report 2013

Section Software

Edition: 2014-03-20

DATA AND KNOWLEDGE REPRESENTATION AND PROCESSING

1. DAHU Project-Team (section vide)	5
2. DREAM Project-Team	6
3. EXMO Project-Team	10
4. GRAPHIK Project-Team	12
5. LINKS Team	14
6. MAGNET Team	15
7. MAIA Project-Team	16
8. OAK Project-Team	18
9. ORPAILLEUR Project-Team	20
10. SMIS Project-Team	24
11. TYREX Team	26
12. WIMMICS Project-Team	29
13. ZENITH Project-Team	38

INTERACTION AND VISUALIZATION

14. ALICE Project-Team	41
15. AVIZ Project-Team	43
16. HYBRID Project-Team	51
17. IMAGINE Project-Team	52
18. IN-SITU Project-Team	55
19. MANAO Team	64
20. MAVERICK Project-Team	66
21. MIMETIC Project-Team	73
22. MINT Project-Team	75
23. POTIOC Team	77
24. REVES Project-Team	78
25. TITANE Team	80

LANGUAGE, SPEECH AND AUDIO

26. ALPAGE Project-Team	81
27. PANAMA Project-Team	87
28. PAROLE Project-Team	88
29. SÉMAGRAMME Project-Team	91

ROBOTICS

30. COPRIN Project-Team	94
31. IMARA Project-Team	95

ROBOTICS AND SMART ENVIRONMENTS

32. E-MOTION Project-Team	97
33. FLOWERS Project-Team	99
34. LAGADIC Project-Team	122

VISION, PERCEPTION AND MULTIMEDIA INTERPRETATION

35. AYIN Team	128
---------------	-----

36. LEAR Project-Team	129
37. MAGRIT Project-Team	131
38. MORPHEO Team	132
39. PERCEPTION Team	135
40. Prima Project-Team	137
41. SIROCCO Project-Team	146
42. STARS Project-Team	147
43. TEXMEX Project-Team	153
44. WILLOW Project-Team	157

DAHU Project-Team (section vide)

DREAM Project-Team

5. Software and Platforms

5.1. Introduction

The pieces of software described in this section are prototypes implemented by members of the project. Any interested person should contact relevant members of the project.

5.2. QTempIntMiner: quantitative temporal sequence mining

QTEMPINTMINER (Quantitative Temporal Interval Miner) is a data mining (cf. 3.2.2) software that implements several algorithms presented in [59] and [3].

The software is mainly implemented in Matlab. A standalone application is now available. It uses the Mixmod toolbox [44] to compute multi-dimensional Gaussian distributions. The main features of QTEMPINTMINER are:

- a tool for generating synthetic noisy sequences of temporal events,
- an implementation of the QTEMPINTMINER, QTIAPRIORI and QTIPREFIXSPAN algorithms,
- a graphical interface that enables the user to generate or import data set and to define the parameters of the algorithm and that displays the extracted temporal patterns.
- a sequence transformer to process long sequences of temporal events. Long sequences are transformed into a database of short temporal sequences that are used as input instances for the available algorithms.

The following website gives many details about the algorithms and provides the latest stable implementation of QTEMPINTMINER: <http://www.irisa.fr/dream/QTempIntMiner/>.

5.3. Sacadeau: qualitative modeling and decision-aid to preserve the water quality from pollutants as herbicides

SACADEAU is an environmental decision software (cf. 4.3) that implements the SACADEAU transfer model. The SACADEAU simulation model couples two qualitative models, a transfer model describing the pesticide transfer through the catchment and a management model describing the farmer decisions. Giving as inputs a climate file, a topological description of a catchment, and a cadastral repartition of the plots, the SACADEAU model simulates the application of herbicides by the farmers on the maize plots, and the transfer of these pollutants through the catchment until the river. The two main simulated processes are the runoff and the leaching. The output of the model simulation is the quantity of herbicides arriving daily to the stream and its concentration at the outlets. The originality of the model is the representation of water and pesticide runoffs with tree structures where leaves and roots are respectively up-streams and down-streams of the catchment.

The software allows the user to see the relationships between these tree structures and the rules learnt from simulations (cf. 3.2.1). A more elaborated version allows to launch simulations and to learn rules on-line. This year, we have developed this new version by enabling access to two recommendation action algorithms. The user can choose different parameters (set of classification rules from which actions will be built, parameters concerning action feasibility, etc) before asking for action recommending process, and then easily visualize the characteristics of situations to improve (polluted ones) compared with the different recommended actions. The software is mainly in Java.

The following website is devoted to the presentation of the SACADEAU: <http://www.irisa.fr/dream/SACADEAU/>. See also [10] for a presentation.

5.4. Ecomata

EcoMata is a tool-box for qualitative modeling and exploring ecosystems and for aiding to design environmental guidelines. We have proposed a new qualitative approach for ecosystem modeling (cf. 4.3) based on timed automata (TA) formalism combined to a high-level query language for exploring scenarios.

To date, EcoMata is dedicated to ecosystems that can be modeled as a collection of species (prey-predator systems) under various human pressures and submitted to environmental disturbances. It has two main parts: the Network Editor and the Query Launcher. The Network Editor let a stakeholder describe the trophic food web in a graphical way (the species icons and interactions between them). Only few ecological parameters are required and the user can save species in a library. The number of qualitative biomass levels is set as desired. An efficient algorithm generates automatically the network of timed automata. EcoMata provides also a dedicated window to help the user to define different fishing pressures, a nice way being by using chronograms. In the Query Launcher, the user selects the kind of query and the needed parameters (for example the species biomass levels to define a situation). Results are provided in a control panel or in files that can be exploited later. Several additional features are proposed in EcoMata: building a species library, import/export of ecosystem model, batch processing for long queries, etc. EcoMata is developed in Java (Swing for the GUI) and the model-checker called for the timed properties verification is UPPAAL.

The following website is devoted to the presentation of ECOMATA: <http://oban.agrocampus-ouest.fr:8080/ecomata>.

5.5. Paturmata

Paturmata is a tool-box for qualitative modeling and exploring agrosystems, specifically management of herd based on pasture [6]. The system is modelled using a hierarchical hybrid model described in timed automata formalism.

In PaturMata software, users can create a pasture system description by entering herds and plots information. For each herd, the only parameter is the number of animals. For each plot, users should enter the surface, the density, the herb height, the distance to the milking shed, a herb growth profile and an accessibility degree .

Users then specify pasturing and fertilization strategies. Finally, users can launch a pasture execution. PaturMata displays the results and a detailed trace of pasture. Users can launch a batch of different strategies and compare the results in order to find the best pasture strategy.

PaturMata is developed in Java (Swing for the GUI) and the model-checker that is called for the timed properties verification is UPPAAL.

Another feature which will be soon added to PaturMata is strategy synthesis. Users choose a pasture configuration or a type of pasture configuration and PaturMata proposes the best pasture and fertilization strategy in order to minimize the pasture procedure cost and use of nitrogen fertilizer.

5.6. ManageYourself

ManageYourself is a collaborative project between Dream and the Telelogos company aiming at monitoring smartphones from a stream of observations made on the smartphone state (cf. 3.2.3).

Today's smartphones are able to perform calls, as well as to realize much more complex activities. They are small computers. But as in computers, the set of applications embedded on the smartphone can lead to problems. The aim of the project ManageYourself is to monitor smartphones in order to avoid problems or to detect problems and to repair them.

The ManageYourself application includes three parts :

- A monitoring part which triggers preventive rules at regular time to insure that the system is working correctly, e.g. *if the memory is full then delete the tmp directory*. This part is always running on the smartphone.

- A reporting part which records regularly the state of the smartphone (the memory state - free vs allocated -, the connection state, which applications are running, etc.). This part also is always running on the smartphone. The current state is stored in a report at regular period and is labeled *normal*. When an application or the system bugs, the current buggy state is stored in a report and is labeled *abnormal*. At regular timestamps, all the reports are sent to a server where the learning process is executed.
- A learning part which learns new bug rules from the report dataset. This part is executed offline on the server. Once the bug rules are learnt, human experts translates them into preventive rules which are downloaded and integrated in the monitoring part of the smartphones.

The following website is devoted to the presentation of MANAGEYOURSELF: <http://www.irisa.fr/dream/ManageYourself/Site/ManageYourself.html>.

5.7. GeoImageRMP: a RapidMiner extension to georeferenced data

RapidMiner is one of the most used frontend for data mining, modelling and analysis. RapidMiner enables the user to design data processing tool chains interactively. A tool chain is a flow chart of processing tools represented by boxes in the interface. This software is easily extendable by designing Plugins. The GeoImageRMP plugin is a plugin dedicated to the design of tool chains to process georeferenced images (raster and vector images) [18]. It is a practical and useful respond to the analytic tasks of georeferenced data. This is the first plugin that is interested in including georeferenced data in RapidMiner and although the only user-friendly tool to create and compare georeferenced data tool chains. It benefits from the large amount of data processing tools that are already implemented in RapidMiner (classification, clustering, frequent pattern mining, etc.). One of the main aims of this plugin is to quickly prototype machine learning tools chain for remote sensing classification task. The GeoImageRMP plugin provide several new processing boxes:

- georeferenced data import/export: create and export dataset that can be processed by standard RapidMiner tool box from/to standard geospatial format (GeoTiff, Shapefiles)
- geospatial sampling method: based on multi-heterogeneous layers of georeferenced data, the sampling method can be transects, random, equidistant, from punctual layer.
- georeferenced data transformation tools: a set of tools dedicated to the manipulation of our new data structures (coordinates, SRS, etc.)
- visualization tool

The following website is devoted to the presentation of GEOIMAGERMP: <http://geoimagermp.gforge.inria.fr/>.

5.8. A plugin for visualizing and editing spatial graphs in QGIS

Spatial graphs are accurate representations of spatial information through spatial objects linked by relationships (spatial or not). This representation is suited to the modeling and analysis of spatial information by computer processing (data mining, search for shortest paths, etc.). While Geographic information System (GIS), such as QGIS, offers the possibility to visualize and manage georeferenced information, the use of spatial graph suffers from the lack of tools to facilitate the construction and integrated visualization.

We developed a QGIS plugin for the visualization and the interactive construction of spatial graphs. QGIS is the most used open source GIS. This plugin introduces a new type of layer: GraphLayer [16]. These new layers can be integrated into any GIS projects. They offer rich functionality for visualization and interactive editing.

5.9. Odisseptale: a software for implementing and evaluating sanitary event detectors in cattle

Odisseptale is a software for implementing disease detectors using monitoring of data provided by sensors placed on calves or cows. Sensors record streams of data such as body temperature, physical activity, feeding behavior, etc. These data are transmitted regularly to a monitoring software that aims to detect if a noticeable

change has occurred on the data streams. Several detectors can be simultaneously active and each contribute to the final decision (detection of a disease). Two kinds of detectors have been implemented: a generic detector based on adaptive CUSUM and a symbolic pattern-based detector. Odisseptale provides also facilities for parameter setting and performance evaluation. This year, the software has been re-implemented in Python for enhanced portability and dissemination.

EXMO Project-Team

5. Software and Platforms

5.1. Alignment API

Participants: Jérôme Euzenat [Correspondent], Jérôme David, Nicolas Guillouet, Armen Inants, Luz Maria Priego-Roche.

We have designed a format for expressing alignments in a uniform way [1]. The goal of this format is to share available alignments on the web. It should help systems using alignments, e.g., mediators, translators, to take advantage of any alignment algorithm and it will help alignment algorithms to be used in many different tasks. This format is expressed in RDF, so it is freely extensible.

The API itself [1] is a JAVA description of tools for accessing the common format. It defines five main interfaces (OntologyNetwork, Alignment, Cell, Relation and Evaluator) and proposes the following services:

- Storing, finding, and sharing alignments;
- Piping matching algorithms (improving an existing alignment);
- Manipulating alignments (thresholding and hardening);
- Generating processing output (transformations, axioms, rules);
- Comparing alignments.

We provide an implementation for this API which can be used for producing transformations, rules or bridge axioms independently from the algorithm which produced the alignment. The proposed implementation features:

- a base implementation of the interfaces with all useful facilities;
- a library of sample matchers;
- a library of renderers (XSLT, RDF, SKOS, SWRL, OWL, C-OWL, SPARQL);
- a library of evaluators (various generalisation of precision/recall, precision/recall graphs);
- a flexible test generation framework which allows for generating evaluation datasets;
- a library of wrappers for several ontology API;
- a parser for the format.

To instantiate the API, it is sufficient to refine the base implementation by implementing the `align()` method. Doing so, the new implementation will benefit from all the services already implemented in the base implementation.

We have developed on top of the Alignment API an Alignment server that can be used by remote clients for matching ontologies and for storing and sharing alignments. It is developed as an extensible platform which allows to plug-in new interfaces. The Alignment server can be accessed through HTML, web service (SOAP and REST) and agent communication interfaces.

The Alignment API is used in the Ontology Alignment Evaluation Initiative data and result processing (§6.1.1). It is also used by more than 30 other teams worldwide.

The Alignment API is freely available since december 2003, under the LGPL licence, at <http://alignapi.gforge.inria.fr>.

5.2. The OntoSim library

Participants: Jérôme David [Correspondent], Jérôme Euzenat.

OntoSim is a library offering similarity and distance measures between ontology entities as well as between ontologies themselves. It materialises our work towards better ontology proximity measures.

There are many reasons for measuring a distance between ontologies. For example, in semantic social networks, when a peer looks for a particular information, it could be more appropriate to send queries to peers having closer ontologies because it will be easier to translate them and it is more likely that such a peer has the information of interest. OntoSim provides a framework for designing various kinds of similarities. In particular, we distinguish similarities in the ontology space from those in the alignment space. The latter ones use available alignments in an ontology network while the former only rely on ontology data. OntoSim is provided with 4 entity measures which can be combined using various aggregation schemes (average linkage, Hausdorff, maximum weight coupling, etc.), 2 kinds of vector space measures (boolean and TF.IDF), and 4 alignment space measures. It also features original comparison methods such as agreement/disagreement measures. In addition, the framework embeds external similarity libraries which can be combined to our own.

OntoSim is based on an ontology interface allowing for using ontology parsed with different APIs. It is written in Java and is available, under the LGPL licence, at <http://ontosim.gforge.inria.fr>.

GRAPHIK Project-Team

5. Software and Platforms

5.1. Cogui

Participants: Alain Gutierrez, Michel Leclère, Marie-Laure Mugnier, Michel Chein, Madalina Croitoru.

Cogui (<http://www.lirmm.fr/cogui>) is a tool for building and verifying knowledge bases. It is a freeware written in Java (version 1.6). Currently, it supports Conceptual Graphs and import/export in RDFS and Datalog+/. This year, the following features have been developed:

- we have introduced the concept of scripted rule which associates more fluidly the editable graphical objects with scripts that perform operations on knowledge graphs. These features have been tested and improved in various projects this year (see *e.g.* Qualinca in Section 8.1 or CTFC in Section 7.2).
- we have implemented an interface for quick and assisted creation of graphs. It is based upon Datalog+/- language and provides tools for automatic completion.
- finally, default conceptual graphs rules were implemented in Cogui. An editing interface is available as well as the operation to find all extensions of a rule set. This feature is required by the CTFC project (see Section 7.2).

5.2. Cogui/Capex

Participants: Alain Gutierrez, Patrice Buche, Awa Diattara, Jérôme Fortin.

Cogui/Capex is a platform for expert knowledge management. It has been developed in order to propose a simple and useful interface to applicative domain experts. This will allow us to validate the integration of our theoretical tools into a real-world application and strengthen GraphIK's involvement in agronomy applications (see the projects with CTFC in Section 7.2 and Panzani in Section 7.3).

5.3. Alaska

Participants: Bruno Paiva Lima Da Silva, Jean-François Baget, Madalina Croitoru.

Alaska (<http://alaska.bplsilva.com/>) is a java library dedicated to the storage and querying of large knowledge bases. It intends to be the foundation layer of our OBDA (Ontology Based Data Access) software developments. It has been built, first as part of a master thesis, and now of Bruno Paiva Lima da Silva' PhD (that will be defended in Jan. 2014).

In Alaska, facts and queries are defined via a generic interface that favors a logical view of these objects. Implementations of this interface allow for the storage of facts w.r.t. different storage paradigms and systems (e.g., relational databases *MySQL* and *Sqlite*; triple stores *Sesame* and graph databases *Neo4J*, *DEX*, *HyperGraphDB* and *OrientDB*). For the time being, we can store 10^7 to 10^8 atoms. In the same way, logical queries can be evaluated through different methods, be it the native querying mechanism of the considered database (e.g. *SPARQL* or *SQL*), or specifically designed algorithms (from a simple backtrack to a full constraint solver based upon *Choco* <http://www.emn.fr/z-info/choco-solver/> for hard problem instances). Note that all these methods provide the same answer set to queries.

This library already allows for testing our OBDA algorithms on large instances. The ADT Quasar (that will start in March 2014) will involve the integration of Alaska with other tools developed in the team (see also Section 5.4), and its improvement from a research library to a distributable tool.

5.4. Tools for Rule-Based Reasoning

Participants: Mélanie König, Michel Leclère, Marie-Laure Mugnier, Swan Rocher, Michaël Thomazo.

Kiabora has been designed to analyze an existential rule base (see Section 6.1) and determine if it allows for finite query answering, i.e., if any conjunctive query evaluated over any fact base while taking this set of rules into account will be answered in a finite time. This year, we fixed some bugs and added some specific options. In addition, a presentation and a demo of *Kiabora* were made at RR 2013 [38].

Besides, the algorithms presented in [38], [37], [41] were implemented and let to experiments. These algorithms are still under development since new improvements have to be integrated.

LINKS Team

4. Software and Platforms

4.1. QuiX-Tool Suite

Participants: Joachim Niehren [correspondant], Denis Debarbieux, Tom Sebastian.

The QuiX-Tool Suite provides tools to process XML streams and documents. The QuiX-Tool Suite is based on early algorithms: query answers are delivered as soon as possible and in all practical cases at the earliest time point. The QuiX-Tool Suite provides an implementation of the main XML standart over streams. XPath, XSLT, XQuery and XProc are W3C standarts while Schematron is an ISO one. The QuiX-Tool suite is developed in the Inria transfer project QuiXProc in cooperation with Innovimax. It includes among the others existing tools such as FXP and QuixPath, along with new tools, namely X-Fun. Both, a free and a professional version are available. The ownership of QuiX-Tool Suite is shared between Inria and Innovimax. The main application of QuiX-Tool Suite is its usage in QuiXProc, an professional implementation of the W3C pipeline language XProc owned by Innovimax.

The QuiXPath language is a large fragment of XPath with full support for the XML data model. The QuiXPath library provides a compiler from QuiXPath to FXP, which is a library for querying XML streams with a fragment of temporal logic.

The X-Fun language is a functional language for defining transformations between XML data trees, while providing shredding instructions. X-Fun can be understood as an extension of Frisch's XStream language with output shredding, while pattern matching is replaced by tree navigation with XPath expressions. The QuiX-Tool suite includes QuiXSLT, which is a compiler from XSLT into a fragment of X-Fun, which can be considered as the core of XSLT. It also provides QuiXSchematron, which is a compiler from Schematron to X-Fun, and QuiXQuery, which is a compiler from XQuery to X-Fun.

See also the web page <https://project.inria.fr/quix-tool-suite/>.

- Version: QuixPath v2.0.0
- Version: X-Fun v0.5.0
- Version: QuiXSLT v0.5.0
- Version: QuiXSchematron v1.0.0

4.2. SmartHal

Participants: Joachim Niehren [correspondant], Antoine Ndione.

SmartHal is a better tool for querying the HAL bibliography database, while is based on Haltool queries. The idea is that a Haltool query returns an XML document that can be queried further. In order to do so, SmartHal provides a new query language. Its queries are conjunctions of Haltool queries (for a list of laboratories or authors) with expressive Boolean queries by which answers of Haltool queries can be refined. These Boolean refinement queries are automatically translated to XQuery and executed by Saxon. A java application for extraction from the command line is available. On top of this, we have build a tool for producing the citation lists for the evaluation report of the LIFL, which can be easily adapter to other Labs.

See also the web page <http://smarthal.lille.inria.fr/>.

- Version: SmartHal v1.0.0

MAGNET Team

5. Software and Platforms

5.1. CoRTex

Participants: Pascal Denis [correspondent], David Chatel.

CoRTex is a LGPL-licensed Python library for Noun Phrase coreference resolution in natural language texts. This library contains implementations of various state-of-the-art coreference resolution algorithms, including those developed in my own research, such as [3]. In addition, it provides a set of APIs and utilities for text pre-processing, reading the main annotation formats (ACE, CoNLL and MUC), and performing evaluation based on the main evaluation metrics (MUC, B-CUBED, and CEAF). As such, CoRTex provides benchmarks for researchers working on coreference resolution, but it is also of interest for developers who want to integrate a coreference resolution within a larger platform. This project is hosted on Inria gforge: <https://gforge.inria.fr/projects/cortex/>.

5.2. JProGraM

Participant: Antonino Freno [correspondent].

JProGraM is a GPL-licensed Java library for machine learning and statistical analysis over graphs and through graphs. Supported models for vectorial data include e.g. Bayesian networks, Markov random fields, Gaussian mixtures, kernel density estimators, and neural networks, whereas random graph tools include small-world networks, preferential-attachment, exponential random graphs, and spectral models (as well as subgraph sampling algorithms). One strong point of the library is the extensive support for continuous random variables. JProGraM integrates implementations for the recent results in [20] and [21]. For more information, see the associated webpage at <http://researchers.lille.inria.fr/~freno/JProGraM.html>.

MAIA Project-Team

5. Software and Platforms

5.1. AA4MM

Participants: Vincent Chevrier [correspondant], Benjamin Camus, Julien Vaubourg.

Laurent Ciarletta (Madyne team, LORIA) is a collaborator and correspondant for this software. Yannick Presse (Madyne team, LORIA) is collaborator for this software.

AA4MM (Agents and Artefacts for Multi-modeling and Multi-simulation) is a framework for coupling existing and heterogeneous models and simulators in order to model and simulate complex systems. The first implementation of the AA4MM meta-model was proposed in Julien Siebert's PhD [65] and written in Java. A newer version with more coupling models is currently submitted to the APP (Agence pour la protection des programmes).

This year, we used this software in a strategic action with EDF R&D in the context of the simulation of smart-grids.

5.2. MASDYNE

Participants: Vincent Chevrier [correspondant], Tomas Navarrete.

This work was undertaken in the PhD Thesis of Julien Siebert, a joint thesis between MAIA and Madyne Team. Laurent Ciarletta (Madyne team, LORIA) has been co-advisor of this PhD and correspondant for this software.

Other contributors to this software were: Tom Leclerc, François Klein, Christophe Torin, Marcel Lamenu, Guillaume Favre and Amir Toly.

MASDYNE (Multi-Agent Simulator of DYnamic Networks usErs) is a multi-agent simulator for modeling and simulating users behaviors in mobile ad hoc network. This software is part of joint work with MADYNES team, on modeling and simulation of ubiquitous networks. It has been updated by Tomas Navarrete with new functionalities for the simulation of scenarii.

5.3. FiatLux

Participant: Nazim Fatès [correspondant].

FiatLux is a discrete dynamical systems simulator that allows the user to experiment with various models (for example 1D and 2D cellular automatas, moving agents on cellular automatas) and to perturb them. Its main feature is to allow users to change the type of updating, for example from a deterministic parallel updating to an asynchronous random updating. FiatLux has a Graphical User Interface and can also be launched in a batch mode for the experiments that require statistics.

In 2013, FiatLux was officially registered by the Agence pour la protection des programmes (APP). A new release is available under the CeCILL licence on the FiatLux website : fiatlux.loria.fr

5.4. Cart-o-matic

Participants: Olivier Simonin [correspondant], François Charpillet, Antoine Bautin, Nicolas Beaufort.

Philippe Lucidarme (Université d'Angers, LISA) is a collaborator and the coordinator of the Cart-o-matic project.

Cart-o-matic is a software platform for (multi-)robot exploration and mapping tasks. It has been developed by Maia members and LISA (Univ. Angers) members during the robotics ANR/DGA Carotte challenge (2009-2012). This platform is composed of three softwares tools which are protected by software copyrights (through the Agence pour la Protection des Programmes): Slam-o-matic a SLAM algorithm developed by LISA members, Plan-o-matic a robot trajectory planning algorithm developed by Maia and LISA members, and Expl-o-matic a distributed multi-agent strategy for multi-robot exploration developed by Maia members (which is based on algorithms proposed in the PhD Thesis of Antoine Bautin). Cf. illustration at [Cart-o-matic](#). The purchase of Cart-o-matic by some robotics companies is underway.

OAK Project-Team

5. Software and Platforms

5.1. Amada

Name: Amada (<https://team.inria.fr/oak/amada/>)

Contact: Jesús Camacho-Rodríguez (jesus.camacho-rodriguez[at]inria.fr)

Other contacts: Ioana Manolescu (ioana.manolescu[at]inria.fr), Dario Colazzo (dario.colazzo[at]dauphine.fr), François Goasdoué (fg[at]irisa.fr)

Presentation: A platform for Web data management in the Amazon cloud.

5.2. FactMinder

Name: FactMinder (<http://tripleo.saclay.inria.fr/xr/demo/>)

Contact: Julien Leblay (julien.leblay[at]inria.fr)

Other contacts: Stamatis Zampetakis (stamatis.zampetakis[at]inria.fr), François Goasdoué (fg[at]irisa.fr), Ioana Manolescu (ioana.manolescu[at]inria.fr)

Presentation: A system for archiving, annotating, and querying semantic-rich Web content.

5.3. Nautilus Analyzer

Name: Nautilus Analyzer (<http://nautilus.saclay.inria.fr/>)

Contact: Melanie Herschel (melanie.herschel[at]lri.fr)

Other contacts: n.a.

Presentation: A tool for analyzing and debugging SQL queries using why-provenance and why-not provenance.

5.4. RDFViewS

Name: RDFViewS (<http://tripleo.saclay.inria.fr/rdfvs/>)

Contact: Konstantinos Karanasos (kkaranasos[at]gmail.com)

Other contacts: François Goasdoué (fg[at]irisa.fr), Julien Leblay (julien.leblay[at]gmail.com), and Ioana Manolescu (ioana.manolescu[at]inria.fr)

Presentation: A storage tuning wizard for RDF applications.

5.5. ViP2P

Name: ViP2P (Views in Peer-to-Peer, <http://vip2p.saclay.inria.fr>)

Contact: Ioana Manolescu (ioana.manolescu[at]inria.fr)

Other contacts: Jesús Camacho-Rodríguez (jesus.camacho-rodriguez[at]inria.fr)

Presentation: A P2P platform for disseminating and querying XML and RDF data in large-scale distributed networks.

5.6. WARG

Name: WARG (<https://team.inria.fr/oak/warg/>)

Contact: Alexandra Roatis (alexandra.roatis[at]lri.fr)

Other contacts: Ioana Manolescu (ioana.manolescu[at]inria.fr), Dario Colazzo (dario.colazzo[at]dauphine.fr), François Goasdoué (fg[at]irisa.fr)

Presentation: A platform for specifying and exploiting warehouses of RDF data.

5.7. XUpOp

Name: XUpOp (XML Update Optimization)

Contact: Dario Colazzo (dario.colazzo[at]dauphine.fr)

Other contacts: Nicole Bidoit (bidoit[at]lri.fr), Mohamed Amine Baazizi (baazizi[at]lri.fr)

Presentation: A general purpose type-based optimizer for XML updates.

5.8. XUpIn

Name: XUpIn (XML Update Independence)

Contact: Federico Ulliana (Federico.Ulliana[at]lri.fr)

Other contacts: Dario Colazzo (colazzo[at]lri.fr), Nicole Bidoit (bidoit[at]lri.fr)

Presentation: An XML query-update independence tester.

5.9. XUpTe

Name: XUpTe (XML Update for Temporal Documents)

Contact: Dario Colazzo (dario.colazzo[at]dauphine.fr)

Other contacts: Nicole Bidoit (bidoit[at]lri.fr), Mohamed-Amine Baazizi (amine.baazizi[at]gmail.com)

Presentation: A type-based optimizer for representing and updating XML temporal data.

5.10. XPUQ

Name: XPUQ (XML Partitioning for Updates and Queries)

Contact: Dario Colazzo (dario.colazzo[at]dauphine.fr)

Other contacts: Nicole Bidoit (bidoit[at]lri.fr)

Presentation: A static analyzer and partitioner for XML queries and updates.

ORPAILLEUR Project-Team

5. Software and Platforms

5.1. Generic Symbolic KDD Systems

5.1.1. *The Coron Platform*

Participants: Jérémie Bourseau [contact person], Aleksey Buzmakov, Victor Codocedo, Adrien Coulet, Amedeo Napoli, Yannick Toussaint.

Keywords: data mining, frequent itemset, closed itemset, generator, association rule, rare itemset

The Coron platform [117], [101] is a KDD toolkit organized around three main components: (1) Coron-base, (2) AssRuleX, and (3) pre- and post-processing modules. The software was registered at the “Agence pour la Protection des Programmes” (APP) and is freely available (see <http://coron.loria.fr>). The Coron-base component includes a complete collection of data mining algorithms for extracting itemsets such as frequent itemsets, closed itemsets, generators and rare itemsets. In this collection we can find APriori, Close, Pascal, Eclat, Charm, and, as well, original algorithms such as ZART, Snow, Touch, and Talky-G. AssRuleX generates different sets of association rules (from itemsets), such as minimal non-redundant association rules, generic basis, and informative basis. In addition, the Coron system supports the whole life-cycle of a data mining task and proposes modules for cleaning the input dataset, and for reducing its size if necessary. The Coron toolkit is developed in Java, is operational, and was already used in several research projects.

5.1.2. *Orion: Skycube Computation Software*

Participant: Chedy Raïssi [contact person].

Keywords: skyline, skycube

This program implements the algorithms described in a research paper published at VLDB 2010 [111]. The software provides a list of four algorithms discussed in the paper in order to compute skycubes. This is the most efficient –in term of space usage and runtime– implementation for skycube computation (see <https://github.com/leander256/Orion>).

5.2. Stochastic systems for knowledge discovery and simulation

5.2.1. *The CarottAge system*

Participants: Florence Le Ber, Jean-François Mari [contact person].

Keywords: Hidden Markov Models, stochastic process

The system CarottAge is based on Hidden Markov Models of second order and provides a non supervised temporal clustering algorithm for data mining and a synthetic representation of temporal and spatial data. CarottAge is currently used by INRA researchers interested in mining the changes in territories related to the loss of biodiversity (projects ANR BiodivAgrim and ACI Ecoger) and/or water contamination. CarottAge is also used for mining hydromorphological data. Actually a comparison was performed with three other algorithms classically used for the delineation of river continuum and CarottAge proved to give very interesting results for that purpose [102].

CarottAge is freely available under GPL license (see <http://www.loria.fr/~jfmari/App/>).

5.2.2. *The ARPeNTAge system*

Participants: Florence Le Ber, Jean-François Mari [contact person].

Keywords: Hidden Markov Models, stochastic process

ARPEntAge¹ (for *Analyse de Régularités dans les Paysages: Environnement, Territoires, Agronomie* is a software based on stochastic models (HMM2 and Markov Field) for analyzing spatio-temporal data-bases [107]. ARPEntAge is built on top of the CarottAge system to fully take into account the spatial dimension of input sequences. It takes as input an array of discrete data in which the columns contain the annual land-uses and the rows are regularly spaced locations of the studied landscape. It performs a Time-Space clustering of a landscape based on its time dynamic Land Uses (LUS). Displaying tools and the generation of Time-dominant shape files have also been defined.

ARPEntAge is freely available (GPL license) and is currently used by INRA researchers interested in mining the changes in territories related to the loss of biodiversity (projects ANR BiodivAgrim and ACI Ecoger) and/or water contamination. In these practical applications, CarottAge and ARPEntAge aim at building a partition –called the hidden partition– in which the inherent noise of the data is withdrawn as much as possible. The estimation of the model parameters is performed by training algorithms based on the Expectation Maximization and Mean Field theories. The ARPEntAge system takes into account: (i) the various shapes of the territories that are not represented by square matrices of pixels, (ii) the use of pixels of different size with composite attributes representing the agricultural pieces and their attributes, (iii) the irregular neighborhood relation between those pixels, (iv) the use of shape files to facilitate the interaction with GIS (geographical information system).

ARPEntAge and CarottAge were used for mining decision rules in a territory showing environmental issues. They provide a way of visualizing the impact of farmers decision rules in the landscape and revealing new extra hidden decision rules [116].

5.3. KDD in Systems Biology

5.3.1. IntelliGO online

The IntelliGO measure computes semantic similarity between terms from a structured vocabulary (Gene Ontology: GO) and uses these values for computing functional similarity between genes annotated by sets of GO terms [83]. The IntelliGO measure is available on line (<http://plateforme-mbi.loria.fr/intelligo/>) to be used evaluation purposes. It is possible to compute the functional similarity between two genes, the intra-set similarity value in a given set of genes, and the inter-set similarity value for two given sets of genes.

5.3.2. WAFObI : KNIME nodes for relational mining of biological data

KNIME (for “Konstanz Information Miner”) is an open-source visual programming environment for data integration, processing, and analysis. KNIME includes a rich library of data manipulation tools (import, export) and several mining algorithms which operate on a single data matrix (decision trees, clustering, frequent itemsets, association rules...). The KNIME platform aims at facilitating the data mining experiment settings as many tests are required for tuning the mining algorithms. The evaluation of the mining results is also an important issue and its configuration is made easier.

Various KNIME nodes were developed for supporting relational data mining using the ALEPH program (<http://www.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.pl>). These nodes include a data preparation node for defining a set of first-order predicates from a set of relation schemes and then a set of facts from the corresponding data tables (learning set). A specific node allows to configure and run the ALEPH program to build a set of rules. Subsequent nodes allow to test the first-order rules on a test set and to perform configurable cross validations.

5.3.3. MOdel-driven Data Integration for Mining (MODIM)

Participants: Marie-Dominique Devignes [contact person], Malika Smaïl-Tabbone.

¹<http://www.loria.fr/~jfmari/App/>

The MODIM software (MOdel-driven Data Integration for Mining) is a user-friendly data integration tool which can be summarized along three functions: (i) building a data model taking into account mining requirements and existing resources; (ii) specifying a workflow for collecting data, leading to the specification of wrappers for populating a target database; (iii) defining views on the data model for identified mining scenarios. A version of the software was declared through Inria APP procedure in December, 2010.

Although MODIM is domain independent, it was used so far for biological data integration in various internal research studies. MODIM was also used for organizing data about non ribosomal peptide syntheses. The sources can be downloaded at <https://gforge.inria.fr/projects/modim/>.

5.4. Knowledge-Based Systems and Semantic Web Systems

5.4.1. *The Kasimir System for Decision Knowledge Management*

Participants: Nicolas Jay, Jean Lieber [contact person], Amedeo Napoli, Thomas Meilender.

Keywords: classification-based reasoning, case-based reasoning, decision knowledge management, knowledge edition, knowledge base maintenance, semantic portal

The objective of the Kasimir system is decision support and knowledge management for the treatment of cancer. A number of modules have been developed within the Kasimir system for editing treatment protocols, visualization, and maintenance. Kasimir is developed within a semantic portal, based on OWL. KatexOWL (Kasimir Toolkit for Exploiting OWL Ontologies, <http://katexowl.loria.fr>) is developed in a generic way and is applied to Kasimir. In particular, the user interface EdHibou of KatexOWL is used for querying the protocols represented within the Kasimir system (see [17] where an extension of Kasimir for multi-viewpoint case-based reasoning is presented).

Cabamaka (case base mining for adaptation knowledge acquisition) is a module of the Kasimir system. This system performs case base mining for adaptation knowledge acquisition and provides information units to be used for building adaptation rules. Actually, the mining process in Cabamaka is based on a frequent close itemset extraction module from the Coron platform (see §5.1.1).

The Oncologik system [12] is a collaborative editing tool aiming at facilitating the management of medical guidelines (<http://www.oncologik.fr/>). Based on a semantic wiki, it allows the acquisition of formalized decision knowledge. Oncologik also includes a graphical decision tree editor called KcatoS.

5.4.2. *Taaable: a system for retrieving and creating new cooking recipes by adaptation*

Participants: Valmi Dufour-Lussier, Emmanuelle Gaillard, Laura Infante Blanco, Florence Le Ber, Jean Lieber, Amedeo Napoli, Emmanuel Nauer [contact person].

Keywords: knowledge acquisition, ontology engineering, semantic annotation, case-based reasoning, hierarchical classification, text mining

Taaable [69] is a system whose objectives are to retrieve textual cooking recipes and to adapt these retrieved recipes whenever needed. Suppose that someone is looking for a “leek pie” but has only an “onion pie” recipe: how can the onion pie recipe be adapted?

The Taaable system combines principles, methods, and technologies such as case-based reasoning (CBR), ontology engineering, text mining, text annotation, knowledge representation, and hierarchical classification. Ontologies for representing knowledge about the cooking domain, and a terminological base for binding texts and ontology concepts, were built from textual web resources. These resources are used by an annotation process for building a formal representation of textual recipes. A CBR engine considers each recipe as a case, and uses domain knowledge for reasoning, especially for adapting an existing recipe w.r.t. constraints provided by the user, holding on ingredients and dish types.

The Taaable system is available on line since 2008 at <http://taaable.fr>. A new version of Taaable was implemented using Tuurbine, a generic ontology-guided CBR engine based on semantic web technologies (see Section 5.4.3). BeGood (see Section 5.4.4), a generic system for managing non-regression tests on knowledge bases, is also plugged for acquiring test sets. When the Taaable system returns answers to a query, the user may evaluate the relevance of the answers. Currently, user feedback is collected using BeGood and will be used in the future to run tests when the knowledge exploited by the CBR system evolves. The objective is to ensure that the knowledge base evolution does not affect the quality of answers given by the CBR system.

5.4.3. *Tuurbine: a generic ontology guided case-based inference engine*

Participants: Laura Infante Blanco, Jean Lieber, Emmanuel Nauer [contact person].

Keywords: case-based reasoning, inference engine, knowledge representation, ontology engineering, semantic web

The experience acquired since 5 years with the Taaable system conducted to the creation of a generic case-based reasoning system, whose reasoning procedure is based on a domain ontology. This new system, called Tuurbine (<http://tuurbine.loria.fr/>), takes into account the retrieval step, the case base organization, but also an adaptation procedure which is not addressed by other generic case-based reasoning tools. Moreover, Tuurbine is built over semantic web standards allowing to be connected to the web of data. The domain knowledge is represented in an RDF store, which can be interfaced with a semantic wiki, for collaborative edition and management of the knowledge involved in the reasoning system (cases, ontology, adaptation rules). The development of Tuurbine was supported by an Inria ADT funding until October 2013.

5.4.4. *BeGood: a generic system for managing non-regression tests on knowledge-bases*

Participants: Laura Infante Blanco, Emmanuel Nauer [contact person].

Keywords: tests, non-regression, knowledge evolution

BeGood [67] is a system allowing to define test plans, independent of any application domain, and usable for testing any system answering queries by providing results in the form of sets of strings. BeGood provides all the features usually found in test systems, such as tests, associated queries, assertions, and expected result sets, test plans (sets of tests) and test reports. The system is able to evaluate the impact of a system modification by running again test plans and by evaluating the assertions which define whether a test fails or succeeds. The main components of BeGood are (1) the “test database” that stores every test artifacts, (2) the “remote query evaluator” which evaluates test queries, (3) the “assertion engine” which evaluates assertions over the expected and effective query result sets. and finally (4) the “REST API” which offers the test functionalities as web services.

BeGood is available under a AGPL license on github². BeGood is used to manage the non-regression of the Taaable system (see Section 5.4.2) when the knowledge base used by the CBR system is modified.

5.4.5. *Revisor: a library of revision operators and revision-based adaptation operators*

Participants: Valmi Dufour-Lussier, Alice Hermann, Florence Le Ber, Jean Lieber [contact person], Emmanuel Nauer, Gabin Personeni.

Keywords: belief revision, adaptation, revision-based adaptation, case-based reasoning, inference engines, knowledge representation

Revisor is a library of inference engines dedicated to belief revision and to revision-based adaptation for case-based reasoning [60]. It is open source, under a GPL license and available on the web (<http://revisor.loria.fr/>). It gathers several engines developed during the previous years, for various knowledge representation formalisms (propositional logic—with or without the use of adaptation knowledge [65]—conjunction of linear constraints, and qualitative algebras [3]). Some of these engines are already used in the Taaable system. Current developments on Revisor aim at defining new engines in other formalisms.

²<https://github.com/kolflow/begood>

SMIS Project-Team

5. Software and Platforms

5.1. Introduction

In our research domain, developing software prototypes is mandatory to validate research solutions and is an important vector for publications, demonstrations at conferences and exhibitions as well as for cooperations with industry. This prototyping task is however difficult because it requires specialized hardware platforms (e.g., new generations of smart tokens), themselves sometimes at an early stage of development.

For a decade, we have developed successive prototypes addressing different application domains, introducing different technical challenges and relying on different hardware platforms. PicoDBMS was our first attempt to design a full-fledged DBMS embedded in a smart card [39] [27]. Chip-Secured Data Access (C-SDA) embedded a reduced SQL query engine and access right controller in a secure chip and acted as an incorruptible mediator between a client and an untrusted server hosting encrypted data [34]. Chip-Secured XML Access (C-SXA) was an XML-based access rights controller embedded in a smart card [35]. Prototypes of C-SXA have been the recipient of the e-gate open 2004 Silver Award and SIMagine 2005 Gold award, two renowned international software contests. The next subsections detail the two prototypes we are focusing on today.

5.2. PlugDB engine

Participants: Nicolas Anciaux [correspondent], Luc Bouganim, Philippe Pucheral, Shaoyi Yin, Yanli Guo, Lionel Le Folgoc, Alexei Troussov.

More than a stand-alone prototype, PlugDB is part of a complete architecture dedicated to a secure and ubiquitous management of personal data. PlugDB aims at providing an alternative to a systematic centralization of personal data. To meet this objective, the PlugDB architecture lies on a new kind of hardware device called Secure Portable Token (SPT). Roughly speaking, a SPT combines a secure microcontroller (similar to a smart card chip) with a large external Flash memory (Gigabyte sized). The SPT can host data on Flash (e.g., a personal folder) and safely run code embedded in the secure microcontroller. PlugDB engine is the cornerstone of this embedded code. PlugDB engine manages the database on Flash (tackling the peculiarities of NAND Flash storage), enforces the access control policy defined on this database, protects the data at rest against piracy and tampering, executes queries (tackling low RAM constraint) and ensures transaction atomicity. Part of the on-board data can be replicated on a server (then synchronized) and shared among a restricted circle of trusted parties through crypto-protected interactions. PlugDB engine has been registered at APP (Agence de Protection des Programmes) in 2009 [28] and a new version is registered each year. The underlying Flash-based indexing system has also been patented by Inria and Gemalto [40]. It has been demonstrated in a dozen of national and international events including JavaOne and SIGMOD. It is being experimented in the field to implement a secure and portable medical-social folder helping the coordination of medical care and social services provided at home to dependent people. The next step in our agenda is to put this software in open-source so that students and communities of developers can complement it and develop innovative privacy-by-design applications. In 2012, we have ported PlugDB-engine on a new hardware platform to 1) become completely independent from Gemalto, 2) have a plug-and-play implementation on Android, 3) serve as a basement to port it on other custom hardware implementations. We have already discussed with hardware companies located in "Ile-de-France" to produce new hardware tokens to host future versions of PlugDB-engine. Link: http://www-smis.inria.fr/_DMSP/home.php.

5.3. uFLIP Benchmark

Participants: Luc Bouganim [correspondent], Philippe Bonnet, Bjorn Jónsson, Lionel Le Folgoc.

It is amazingly easy to produce meaningless results when measuring flash devices, partly because of the peculiarity of flash memory, but primarily because their behavior is determined by layers of complex, proprietary, and undocumented software and hardware. uFLIP is a component benchmark for measuring the response time distribution of flash IO patterns, defined as the distribution of IOs in space and time. uFLIP includes a benchmarking methodology which takes into account the particular characteristics of flash devices. The source code of uFLIP, available on the web (700 downloads, 4000 distinct visitors), was registered at APP in 2009 [32]. It has been demonstrated at SIGMOD.

Link: <http://www.uflip.org>.

TYREX Team

5. Software and Platforms

5.1. XML Reasoning Solver

Participants: Pierre Genevès, Nabil Layaida, Nils Gesbert, Manh-Toan Nguyen.

The **XML Reasoning Solver** is a tool for the static analysis of XPath queries and XML schemas based on the latest theoretical advances [9]. It allows automated verification of properties that are expressed as logical formulas over trees. A logical formula may for instance express structural constraints or navigation properties (like e.g. path existence and node selection) in finite trees.

The tool can solve many fundamental XML problems such as satisfiability of XPath expressions in the presence of XML schemas, containment and equivalence of XPath expressions, and many other problems that can be formulated with XPath expressions and schemas (DTDs, XML Schemas, Relax-NG).

The system is implemented in Java and uses symbolic techniques (binary decision diagrams) in order to enhance its performance. It is capable of comparing path expressions in the presence of real-world DTDs (such as the W3C SMIL and XHTML language recommendations, for instance). The cost ranges from a few milliseconds, for comparison of XPath queries without tree types, to several seconds for queries under very large and heavily recursive type constraints such as the XHTML DTD. These measurements shed light for the first time on the cost of solving static analysis problems in practice. Furthermore, the analyzer generates XML counter-examples that allow program defects to be reproduced independently from the analyzer.

5.1.1. Extensions for CSS

We have introduced the first system capable of statically verifying properties of a given cascading style sheet (CSS) over the whole set of documents to which this stylesheet applies [8]. The system is composed of a set of parsers for reading the CSS and schema files (XML Schema, Relax NG, or DTD) together with a text file corresponding to the problem description as a logical formula. We have developed a compiler that translates CSS files into their logical representations. Then, the solver takes the overall problem formulation and checks it for satisfiability.

5.1.2. XQuery IDE

We have started the development of an XQuery IDE with a web interface. This prototype integrates static analyses performed by the solver inside a development environment suited for XQuery programmers.

5.2. ClaireCourseMaker Library

Participants: Nicolas Hairon, Cécile Roisin.

The goal of the ClaireCourseMaker is to provide straightforward editing tools for structuring, annotating and timeline-based authoring of continuous content such as audio or video. Even if it can be used for any content, it is mainly devoted to synchronize pedagogical material (video, slides, chaptering, etc.) in order to provide rich media online courses à la MOOC. The underlying technology is standard-based and uses the open source JavaScript Popcorn library and Popcorn Maker web application by Mozilla.

The result is a wysiwyg web-based authoring tool which benefits from all the generic features of Popcorn and the specific services that cope with chaptering and synchronization needs.

ClaireCourseMaker is the direct follow-up tool of the Timesheet library which is a cross-browser JavaScript implementation for scheduling the dynamic behavior of HTML5 content that can be described with declarative SMIL markup (**SMIL Timing and Synchronization**, **SMIL Timesheets**).

ClaireCourseMaker is developed in collaboration with the OpenClassrooms company in the context of the Claire project (see section 7.1.1).

5.3. Mobile Audio Language

Participants: Yohan Lasorsa, Jacques Lemordant.

5.3.1. MAUDL library

The MAUDL library (Mobile AUDIO Language) is an evolution of the ARIA library whose primary target was games on mobile devices.

Augmented Reality Audio applications use sound objects to create a soundscape. A sound object is a time structure of audio chunks whose duration is on the time scale of 100 ms to several seconds. These sound objects have heterogeneous and time-varying properties. In order to describe Interactive Audio (IA) contents, we created MAUDL, an XML language inspired by iXMF that is well adapted to the design of dynamic soundtracks for navigation systems.

MAUDL prevents audio information overwhelming through categorization at the declarative level and the use of priority queues at the execution level. This takes account of speed when walking, and of rapid hand gestures when interrogating the environment, for example. MAUDL can be used as an authoring time interchange file format for interactive mobile applications or as a runtime file format that is actually loaded through the web and played directly in the device. MAUDL is a cue-oriented interactive audio system, where audio services are requested using named events and the system's response to each event is determined by the audio artist.

The library has been implemented in C++ and now supports different mobile operating systems such as Android and iOS. MAUDL has been widely used in the first demonstrator of the VENTURI project consisting of a mobile augmented reality game.

5.3.2. 3D Audio Pointer

A virtual 3D audio pointer provides an intuitive guide to the user of a mobile navigation application, reducing the need for cognitive work when compared to vocal instructions. We have built such a pointer using the MAUDL language. It gives the user the azimuth using HRTF spatialized audio cues, with additional hints taking the form of variations in the sound used. It allows superposing other kinds of audio contents, such as voice while the pointer is active, to indicate distance for example. This audio object is suitable for different sorts of navigation systems, such as POI browsers, self-guided audio tours, or applications for following predefined routes.

As the 3D audio pointer is based on MAUDL this technology is also available for both platforms, iOS and Android. It has been used by **Metaio** for the micro-navigation which is part of the second VENTURI demonstrator. The use case was to help a visually impaired person to find a box on a shelf with computer vision and 3D audio rendering.

5.4. PDRTrack

Participants: Jacques Lemordant, Mathieu Razafimahazo.

PDRTrack is a localization utility running on iOS or Android smartphones used for recording and playing data sets (accelerometer, gyroscope, barometer and magnetometer values) to find out the effect of different pedometer and map matching parameters and their result on localization accuracy. This application uses the PDR library, written in C++ and developed by the team, which provides the user's location in real time based on the interpretation of sensor readings. Three main modules have been designed to build this localization system:

- a pedometer that estimates the distance the user has walked and his speed
- a motion manager that enables data set recording and simulation but also the creation of virtual sensors or filters (e.g gyroscope drift compensation, linear acceleration, altimeter)
- a map-matching algorithm that provides a new location based on a given OpenStreetMap file description and the current user's trajectory

The PDR library has been shared to the VENTURI consortium for the first part of the second year demonstrator: guiding a visually impaired person from Fondazione Bruno Kessler's bus stop to the building entrance. Others partners have used this localization system for retrieving a scale factor needed for the computer vision part (i.e SLAM).

5.5. Interactive eXtensible Engine (IXE)

Participants: Yohan Lasorsa, Jacques Lemordant, David Liodenot, Thibaud Michel, Mathieu Razafimahazo.

GPS navigation systems, when used in an urban environment, are limited in precision and can only give instructions at the level of the street and not of the pavement. GPS is also limited to outdoor navigation and requires some delicate transitioning system when switching to another positioning system to perform indoor navigation.

IXE is an open source urban pedestrian navigation system based on Inertial Measurement Units (IMU) and running on mobile phones with onboard geographic data and a routing engine. With IXE, the distinction between indoor and outdoor is blurred as an IMU-based location engine can run indoor and outdoor. IXE allows augmented reality queries on customized embedded geographical data. Queries on route nodes or POIs, on ways and relations are predefined for efficiency and quality of information.

Following the web paradigm, IXE is a browser for XML documents describing navigation networks: by using the micro-format concept, one can define inside OpenStreetMap a complex format for pedestrian navigation networks allowing navigation at the level of pavements or corridors. The big advantage of doing this instead of defining new XML languages is that we can use the standard OpenStreetMap editor JOSM to create navigation networks in a short amount of time.

The purpose of the IXE browser is to read these OSM documents and to generate from them visible or audible navigation information. IXE works on any mobile phone running under iOS or Android. Its heart is composed of three engines, one for dead-reckoning navigation, one for interactive audio and the last one for Augmented Reality visual information, allowing quick reconfiguration for extremely varied applications.

IXE can be used for accessible navigation allowing independent living for people with disabilities.

IXE Android is an enhanced version of our iOS navigation demonstrator. It uses our latest work on the localization positioning system such as PDR, GPS, user and NFC. This application is based on predetermined walks described in a XML format extending OpenStreetMap for navigation purpose, everybody can create and share their walks. In order to create a new walk, the author has to follow specifications described in part 6.3.1. We added some functionalities to the open source application Java OpenStreetMap Editor to enhance walk authoring for the IXE app.

- [IXE-iOS](#)
- [IXE-Android](#)

WIMMICS Project-Team

5. Software and Platforms

5.1. Corese

Participants: Olivier Corby [correspondant], Alban Gaignard, Fabien Gandon.

Corese (COnceptual REsource Search Engine) is a Semantic Web Factory. It enables users to load and process RDFS schemas, RDF data and query and update the graph base thus created by using the SPARQL 1.1 Query & Update Language (figure 1).

Furthermore, Corese query language integrates original features such as approximate search, extended Property Path, SQL or XPath. It provides a SPARQL based pretty printing language for RDF graphs and a SPARQL based Inference Rule Language for RDF. Corese also provides distributed federated query processing, thanks to a collaboration with Alban Gaignard and Johan Montagnat from CNRS I3S.

Corese is a Semantic Web Factory that enables us to design and develop Semantic Web applications; it is available for download. In the past, Corese received two software development grants (ADT) from Inria and in 2013 we have a new grant for two more years. Corese is registered at the APP and in 2007 we decided to distribute it as open source software under license CeCILL-C.

Corese is used and has been used in more than 60 applications, 24 PhD Thesis and is used for education by several institutions. It has been used in European projects such as Ontorule, Palette, SevenPro, SeaLife and in ANR projects such as Kolflow, Ginseng, Neurolog, VIP, ISICIL, e-WOK Hub. Corese is the Semantic Web engine of Discovery Hub and of the Semantic Web Import Plugin for Gephi visualization.

The work on Corese was published in [2], [1], [3], [4].

Web page: <http://wimmics.inria.fr/corese>

5.2. Semantic Web Import Plugin for Gephi visualization

Participants: Erwan Demairy, Fabien Gandon, Olivier Corby.

The SemanticWebImport² plugin is intended to allow the import of semantic data into Gephi open graph visualization platform (figure 2). Gephi is an interactive visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs. The imported data are obtained by processing a SPARQL request on the semantic data. The data can be accessed following three manners:

1. by accessing local RDF & RDFS files and using the embedded Corese engine to apply the SPARQL request;
2. by accessing a remote REST SPARQL endpoint. In that case, the SPARQL request is applied remotely and the graph is built locally by analyzing the result sent by the endpoint;
3. by accessing a remote SOAP SPARQL endpoint. As for the REST endpoint, the resulting graph is built from the result returned by the endpoint.

The software is released under version 1.0. It has received a development grant (ADT) from Inria.

Web pages: <http://wiki.gephi.org/index.php/SemanticWebImport>
<https://gforge.inria.fr/projects/segviz-public>

5.3. Datalift Linked Open Data Platform

Participants: Luca Costabello, Fabien Gandon, Serena Villata.

The Datalift platform aims at easing and automating publication of raw structured datasets on the Web of data. The platform proposes an extensible architecture and comes with modules enabling: data selection, schema selection and mapping; format and vocabulary conversion; storage, publication querying and access control; interlinking with other sources; visualization. The latest version of the code is maintained on the public forge of Inria³.

²<http://wiki.gephi.org/index.php/SemanticWebImport>

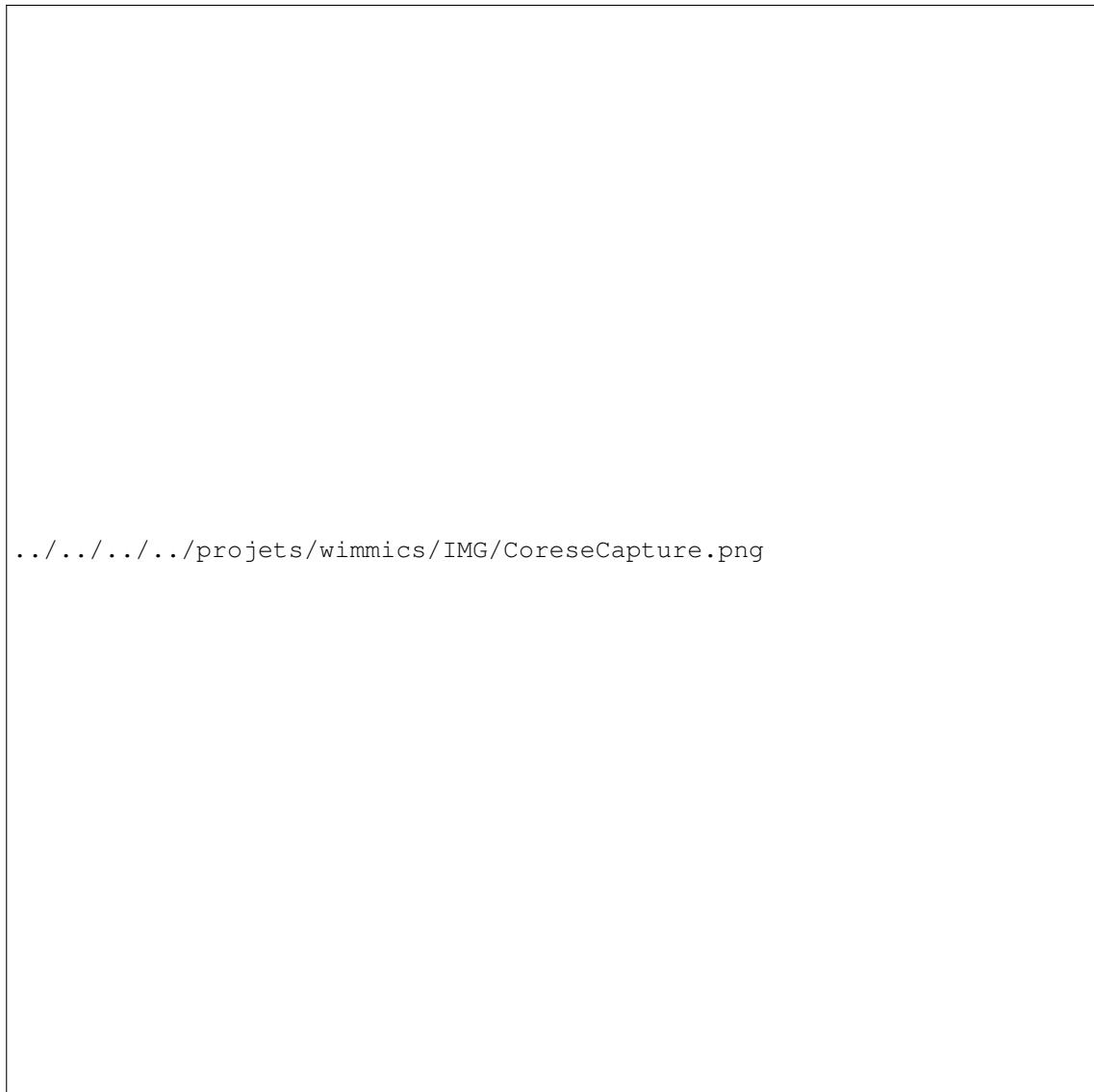


Figure 1. Corese



Figure 2. Gephi

Web page: <http://www.datalift.org>

5.4. Question Answering Wikiframework-based System

Participant: Elena Cabrio.

The QAKiS system (figure 3) implements question answering over DBpedia. QAKiS allows end users to submit a query to an RDF triple store in English and obtain the answer in the same language, hiding the complexity of the non-intuitive formal query languages involved in the resolution process. At the same time, the expressiveness of these standards is exploited to scale to the huge amounts of available semantic data. Its major novelty is to implement a relation-based match for question interpretation, to convert the user question into a query language (e.g. SPARQL). English, French and German DBpedia chapters are the RDF data sets to be queried using a natural language interface.

Web page: <http://www.qakis.org>

5.5. French Chapter of DBpedia

Participants: Julien Cojan, Fabien Gandon.

DBpedia is an international crowd-sourced community effort to extract structured information from Wikipedia and make this information available on the semantic Web as linked open data. The DBpedia triple stores then allow anyone to solve sophisticated queries against Wikipedia extracted data, and to link the different data sets on these data. The French chapter of DBpedia was created and deployed by Wimmics and is now an online running platform providing data to several projects such as: QAKIS, Izipedia, zone47, Sépage, HdA Lab., JocondeLab, etc.

The platform can be found at: <http://www.dbpedia.fr>.

It is part of the Semanticpedia convention: <http://www.semanticpedia.org/>.

5.6. Semantic Wiki

Participants: Pavel Arapov, Michel Buffa.

WikiNEXT is a semantic wiki prototype (figure 4) written in JavaScript, from database to server and client code. It is not in competition with wikis like Semantic Media Wiki, but more a test bed for new ideas. Every wiki page is an application that keeps a Web Socket open with the server, enabling incremental saves or collaborative editions using Google wave like algorithms. Using JavaScript on the whole chain of operations avoids data transformation from/to different formats like in traditional approaches (Objects, JSON/XML, and SQL). WikiNEXT uses JavaScript distributed objects and includes an IDE to write JS applications within wiki pages.

Web page: <http://wikinext.gexsoft.com>

5.7. ISICIL

Participants: Nicolas Delaforge, Fabien Gandon [resp.].

In the context of the ISICIL ANR project, we have developed a Semantic Web server which provides core services to manage simple tagging of resources (internal or from the Web) and to assist the semantic enrichment of the folksonomy of our communities of users. This server's implementation is based on the ISICIL main framework. The tagging model combines already existing ontologies such as SIOC⁴, SCOT, and Newman's Tag Ontology⁵. SRTag, the model of folksonomy enrichment, is based on a named graph mechanism in order to maintain diverging statements made between tags using SKOS (for thesaurus like relation between tags) or SCOT (for spelling variant relations), and is shown in figure 5 .

³<https://gforge.inria.fr/projects/datalift/>

⁴<http://sioc-project.org>

⁵<http://www.holygoat.co.uk/owl/redwood/0.1/tags>



Figure 3. QAKIS

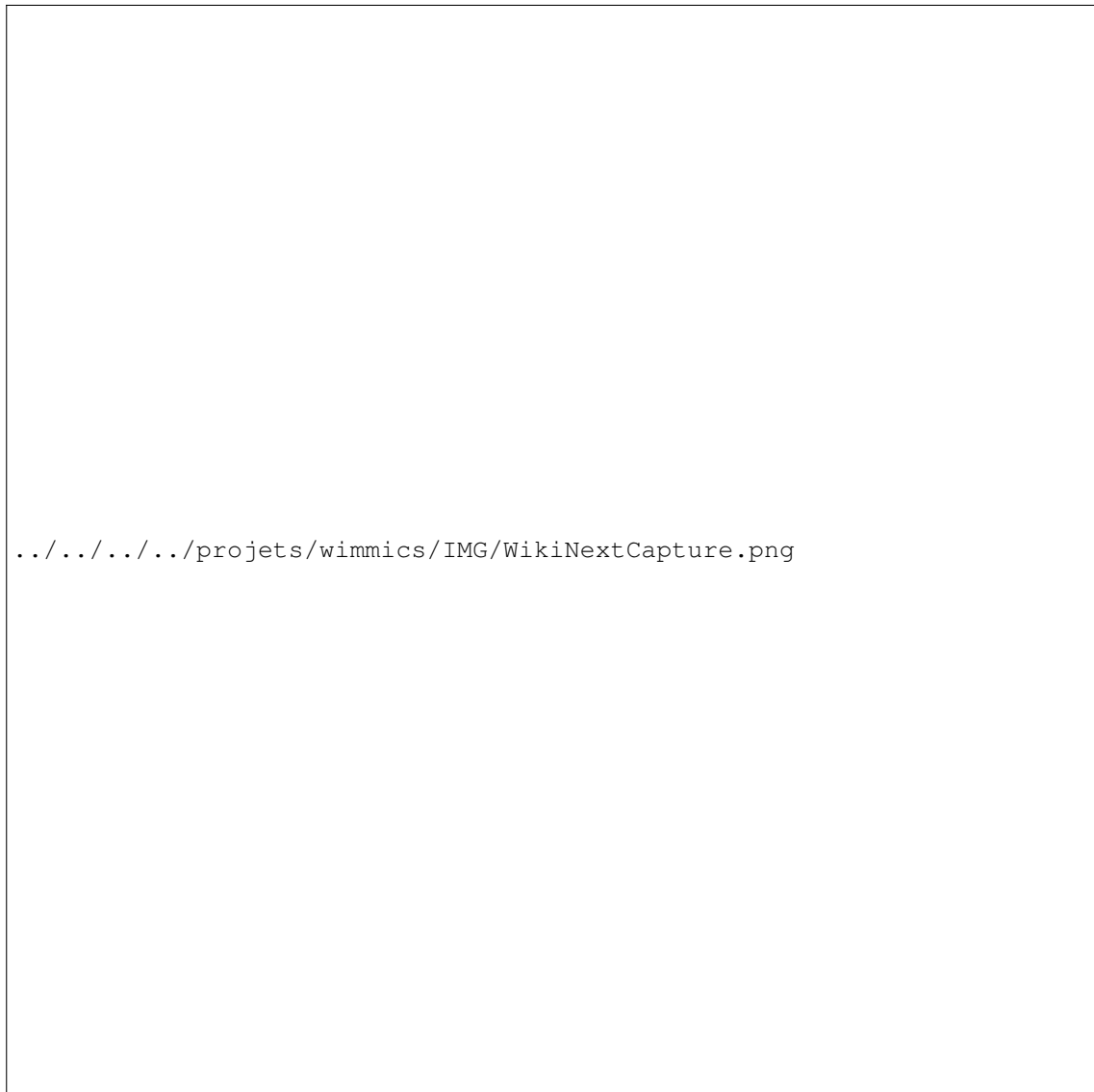


Figure 4. WikiNEXT



Figure 5. Folksonomy enrichment model

Web page: <https://gforge.inria.fr/projects/isicil/>

The code is now being refactored by the company Mnemotix, a SCOOP created as a spin-off of the project: <http://www.mnemotix.com>.

5.8. ZONE-project

Participant: Christophe Desclaux.

ZONE-project provides a new, innovative way to follow news (figure 6). At its core, the system is aggregating news items from various RSS feeds. Using the power of Semantic Web we are able to efficiently tag & annotate each news. Those tags are the basis of filters. Filters allow users to see only news that are relevant. For instance users can retrieve all news containing a tag, or on the contrary never see news containing specific tags. Basically it means that each user can create custom news feeds according to his interests. Though it may be tedious for John Doe to build its own filters, thus it will be possible to exchange filters with other users, or read specific news feeds built by other users. This will enable users to create news group feed focused on specific topics such as technology, health, industry, transport, agriculture, communication, environment, etc. This project won the Inria BoostYourCode 2012 contest which was created in order to promote free & open source software.

Web page: <http://www.zone-project.org/>

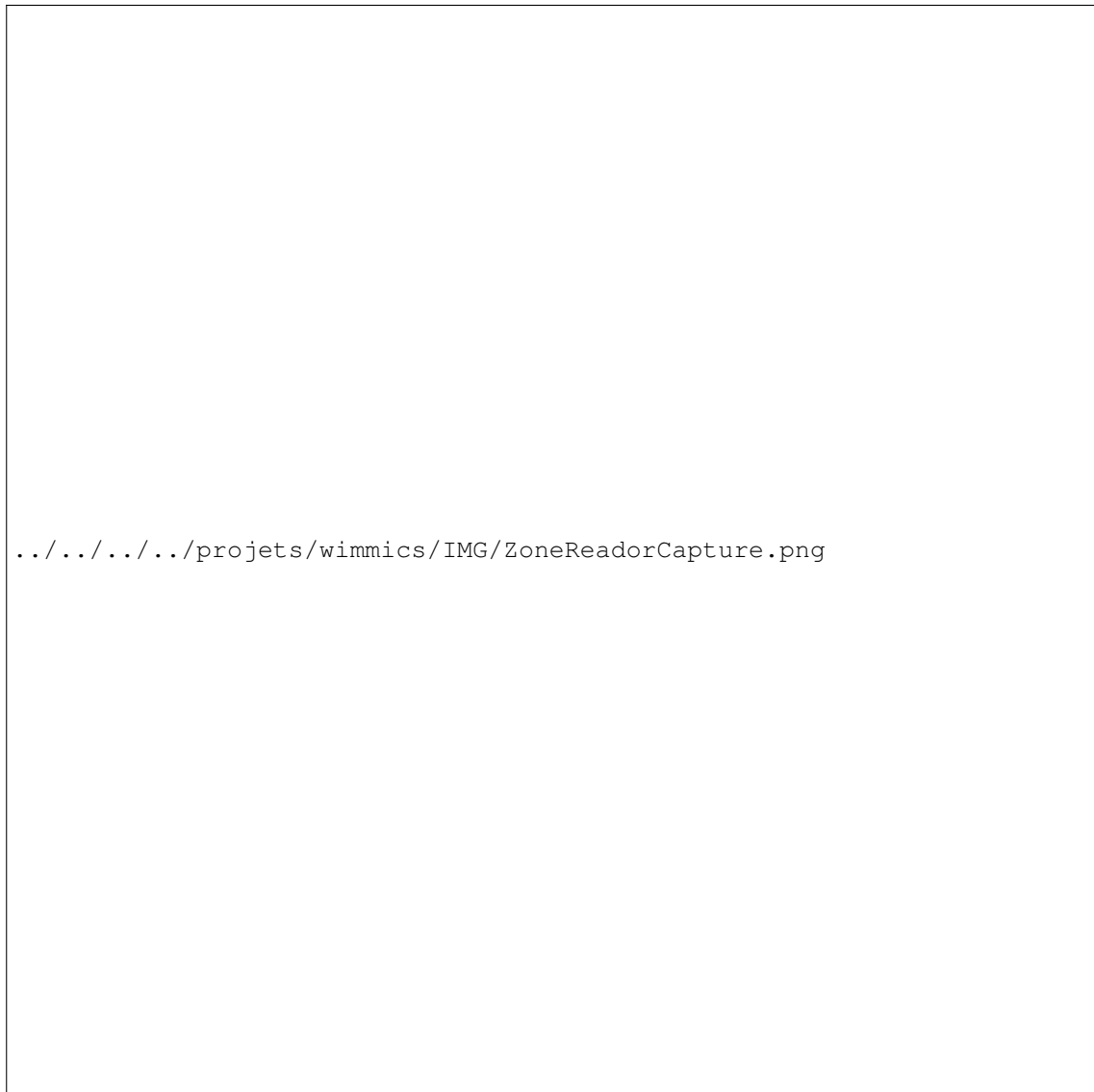


Figure 6. ZoneReader

ZENITH Project-Team

5. Software and Platforms

5.1. WebSmatch (Web Schema Matching)

Participants: Emmanuel Castanier, Rémi Coletta, Patrick Valduriez [contact].

URL: <http://websmatch.gforge.inria.fr/>

In the context of the Action de Développement Technologique (ADT) started in october 2010, WebSmatch is a flexible, open environment for discovering and matching complex schemas from many heterogeneous data sources over the Web. It provides three basic functions: (1) metadata extraction from data sources; (2) schema matching (both 2-way and n-way schema matching), (3) schema clustering to group similar schemas together. WebSmatch is being delivered through Web services, to be used directly by data integrators or other tools, with RIA clients. Implemented in Java, delivered as Open Source Software (under LGPL) and protected by a deposit at APP (Agence de Protection des Programmes). WebSmatch is being used by Datapublica and CIRAD to integrate public data sources.

5.2. SON (Shared-data Overlay Network)

Participants: Esther Pacitti, Didier Parigot [contact], Patrick Valduriez.

URL: <http://www-sop.inria.fr/teams/zenith/SON>

SON is an open source development platform for P2P networks using web services, JXTA and OSGi. SON combines three powerful paradigms: components, SOA and P2P. Components communicate by asynchronous message passing to provide weak coupling between system entities. To scale up and ease deployment, we rely on a decentralized organization based on a DHT for publishing and discovering services or data. In terms of communication, the infrastructure is based on JXTA virtual communication pipes, a technology that has been extensively used within the Grid community. Using SON, the development of a P2P application is done through the design and implementation of a set of components. Each component includes a technical code that provides the component services and a code component that provides the component logic (in Java). The complex aspects of asynchronous distributed programming (technical code) are separated from code components and automatically generated from an abstract description of services (provided or required) for each component by the component generator.

5.3. P2Prec (P2P recommendation service)

Participants: Esther Pacitti [contact], Didier Parigot, Maximilien Servajean.

URL: <http://p2prec.gforge.inria.fr>

P2Prec is a recommendation service for P2P content sharing systems that exploits users social data. To manage users social data, we rely on Friend-Of-A-Friend (FOAF) descriptions. P2Prec has a hybrid P2P architecture to work on top of any P2P content sharing system. It combines efficient DHT indexing to manage the users FOAF files with gossip robustness to disseminate the topics of expertise between friends. P2Prec is implemented in java using SON.

5.4. ProbDB (Probabilistic Database)

Participants: Reza Akbarinia [contact], Patrick Valduriez.

URL: <http://probdb.gforge.inria.fr>

ProbDB is a probabilistic data management system to manage uncertain data on top of relational DBMSs. One of the main features of the prototype is its portability; that means with a minimum effort it can be implemented over any DBMS. In ProbDB, we take advantage of the functionalities provided by almost all DBMSs, particularly the query processing functions. It is implemented in Java on top of PostgreSQL.

5.5. PI@ntNet-mobile

Participants: Vera Bakic, Souheil Selmi, Hervé Goëau, Alexis Joly [contact].

URL: <http://goo.gl/CpSrr3>

PI@ntNet-mobile is an image sharing and retrieval application for the identification of plants built in the continuity of the former web application PI@ntNet-Identify ² (presented in last year activity report). It is developed in the context of the PI@ntNet project that involves four French research organisations (Inria, Cirad, INRA, IRD) and the members of Tela Botanica social network. The key feature of this free app is to help identifying plant species from photographs, through a server-side visual search engine based on several results of ZENITH team on content-based information retrieval. Since its first release in March 2013 on the apple store, the application was downloaded by around 80K users in about 150 countries (between 200 and 2000 active users daily with peaks occurring during the week-ends). The collaborative training set that allows the content-based identification is continuously enriched by the users of the application and the members of Tela Botanica social network. At the time of writing, it includes about 80K images covering more than 3500 French plant species about 2/3 of the whole French flora (this is actually the widest identification tool built anytime).

5.6. PI@ntNet-DataManager

Participants: Mathias Chouet [contact], Alexis Joly.

PI@ntNet-DataManager ³ is a software dedicated to managing and sharing distributed heterogeneous botanical data. It is developed jointly by Zenith, the AMAP UMR team (CIRAD) and Telabotanica non profit organization. It allows scientists to define data structures dedicated to their own datasets, and share parts of their structures and data with collaborators in a decentralized way. PI@ntNet DataManager offers innovative features like partial or complete P2P synchronization between distant databases (master-master), and a user friendly data structure editor. It also provides full text search, querying, CSV import/export, SQL export, image management, and geolocation. DataManager is built on NoSQL technology (CouchDB database), Javascript (Node.js), HTML5 and CSS3, and may be deployed on a server or run on a local machine (standalone version for Linux, Windows, Mac). It is being used by researchers and engineers of the PI@ntNet Project (CIRAD, INRA, Inria, IRD, Tela-Botanica) to manage taxonomical referentials, herbarium data and geolocated plant observations.

5.7. SnoopIm

Participants: Julien Champ [contact], Alexis Joly, Pierre Letessier.

URL: <http://otmedia.lirmm.fr/>

SnoopIm is a content-based search engine allowing to discover and retrieve small visual patterns or objects in large collections of pictures and to derive statistics from them (frequency, visual cover, size variations, etc.). It is implemented in Javascript on top of a C++ library developed in collaboration with INA ⁴. The software is used at INA by archivists and sociologists in the context of the Transmedia Observatory project. It is also being experimented in several contexts including a logo retrieval application set up in collaboration with the French Press Agency, an experimental plant identification tool mixing textual and visual information retrieval (in the context of the PI@ntNet project) and a research project on high-throughput analysis of root architecture images.

²<http://identify.plantnet-project.org>

³<http://data.plantnet-project.org/>

⁴<http://www.ina-sup.com/>

5.8. SciFloware

Participants: Dimitri Dupuis, Didier Parigot [contact], Patrick Valduriez.

URL: <http://www-sop.inria.fr/members/Didier.Parigot/pmwiki/Scifloware>

SciFloware is an action of technology development (ADT Inria) with the goal of developing a middleware for the execution of scientific workflows in a distributed and parallel way. It capitalizes on our experience with SON and an innovative algebraic approach to the management of scientific workflows. SciFloware provides a development environment and a runtime environment for scientific workflows, interoperable with existing systems. We will validate SciFloware with workflows for analyzing biological data provided by our partners CIRAD, INRA and IRD.

ALICE Project-Team

4. Software and Platforms

4.1. Vorpaline

Participants: Dobrina Boltcheva, Bruno Lévy, Thierry Valentin.

Vorpaline is an automatic surfacic and volumetric mesh generation software, distributed with a commercial license. Vorpaline is based on the main scientific results stemming from projets GoodShape and VORPALINE, funded by the European Research Council, about optimal quantization, centroidal Voronoi diagrams and fast/parallel computation of Voronoi diagrams in high-dimension space. The current version (1.0) provides functionalities such as isotropic/adaptive/anisotropic surface re-meshing, tolerant surface re-meshing, mesh repair and mesh decimation. Next versions will provide functionalities such as constrained surface meshing (2.0), quad-dominant surface meshing (3.0) and hex-dominant volume meshing (4.0).

4.2. IceSL

Participants: Jérémie Dumas, Jean Hergel, Sylvain Lefebvre.

In the new software **IceSL**, we propose to exploit recent advances in GPU and Computer Graphics to accelerate the slicing process of objects modelled via a CSG¹ language. Our target are open source low cost *fused deposition modeling* printers such as RepRaps.

Our approach first inputs a CSG description of a scene which can be composed of both meshes and analytic primitives. During display and slicing the CSG model is converted on the fly into an intermediate representation enabling fast processing on the GPU. Slices can be quickly extracted, and the tool path is prepared through image erosion. The interactive preview of the final geometry uses the exact same code path as the slicer, providing an immediate, accurate visual feedback.

IceSL is the recipient software for our ERC research project “ShapeForge”, led by Sylvain Lefebvre.

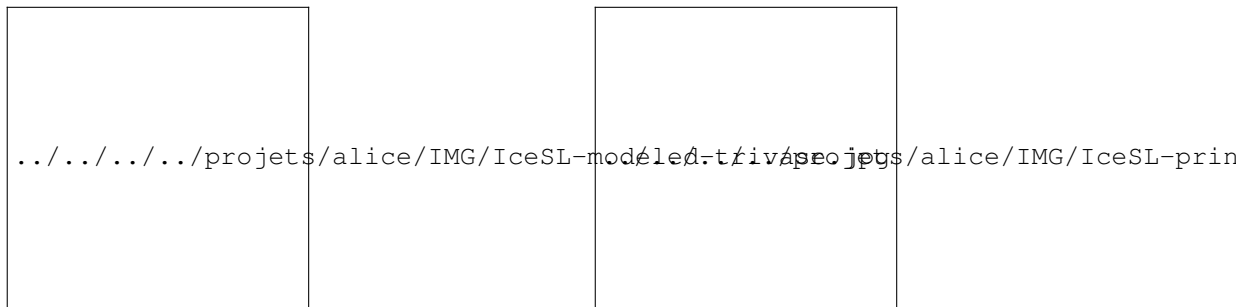


Figure 1. Left. A two-colored vase is modeled in IceSL. Right. An early printed result.

4.3. Graphite

Participants: Dobrina Boltcheva, Samuel Hornus, Bruno Lévy, David Lopez, Romain Merland, Jeanne Pellerin, Nicolas Ray.

¹ Constructive Solid Geometry

Graphite is a research platform for computer graphics, 3D modeling and numerical geometry. It comprises all the main research results of our “geometry processing” group. Data structures for cellular complexes, parameterization, multi-resolution analysis and numerical optimization are the main features of the software. Graphite is publicly available since October 2003. It is hosted by Inria GForge since September 2008. Graphite is one of the common software platforms used in the frame of the European Network of Excellence **AIMShape**.

Graphite and its research-plugins are actively developed and extended. The latest version was released on January 2nd, 2014 and has been downloaded 135 times as of January 29.

4.4. GraphiteLifeExplorer

Participant: Samuel Hornus.

GLE is a 3D modeler, developed as a plugin of Graphite, dedicated to molecular biology. It is developed in cooperation with the Fourmentin Guilbert foundation and has recently been renamed "GraphiteLifeExplorer". Biologists need simple spatial modeling tools to help in understanding the role of the relative position of objects in the functioning of the cell. In this context, we develop a tool for easy DNA modeling. The tool generates DNA along any user-given curve, open or closed, allows fine-tuning of atoms position and, most importantly, exports to PDB (the Protein Data Bank file format).

The development of GLE is currently on hold, but it is still downloaded (freely) about twice a day (1600 downloads to date). A paper describing it was published in the broad journal PLOS One [12].

4.5. OpenNL - Open Numerical Library

Participants: Bruno Lévy, Nicolas Ray, Rhaleb Zayer.

OpenNL is a standalone library for numerical optimization, especially well-suited to mesh processing. The API is inspired by the graphics API OpenGL, this makes the learning curve easy for computer graphics practitioners. The included demo program implements our LSCM [5] mesh unwrapping method. It was integrated in **Blender** by Brecht Van Lommel and others to create automatic texture mapping methods. OpenNL is extended with two specialized modules :

- **CGAL parameterization package:** this software library, developed in cooperation with Pierre Alliez and Laurent Saboret, is a **CGAL** package for mesh parameterization.
- **Concurrent Number Cruncher:** this software library extends OpenNL with parallel computing on the GPU, implemented using the CUDA API.

4.6. LibSL

Participants: Anass Lasram, Sylvain Lefebvre.

LibSL is a Simple library for graphics. Sylvain Lefebvre continued development of the LibSL graphics library (under CeCill-C licence, filed at the APP). LibSL is a toolbox for rapid prototyping of computer graphics algorithms, under both OpenGL, DirectX 9/10, Windows and Linux. The library is actively used in both the REVES / Inria Sophia-Antipolis Méditerranée and the ALICE / Inria Nancy Grand-Est teams.

AVIZ Project-Team

5. Software and Platforms

5.1. Graph Cuisine

Participants: Évelyne Lutton [correspondant], Benjamin Bach, André Spritzer, Jean-Daniel Fekete.



Figure 1. GraphCuisine interface showing one real graph (right), the measures extracted from it (bottom circles), several graphs with similar measures (left) and one of them selected (middle).

GraphCuisine lets users steer an Evolutionary Algorithm (EA) to create random graphs that match user-specified measures. Generating random graphs with particular characteristics is crucial for evaluating graph algorithms, layouts and visualization techniques. Current random graph generators provide limited control of the final characteristics of the graphs they generate. The situation is even harder when one wants to generate random graphs similar to a given one, all-in-all leading to a long iterative process that involves several steps of random graph generation, parameter changes, and visual inspection. Our system follows an approach based on interactive evolutionary computation. Fitting generator parameters to create graphs with pre-defined measures is an optimization problem, while assessing the quality of the resulting graphs often involves human subjective judgment. GraphCuisine has been proved to be able to generate graphs that mimic a given real-world network. <http://www.aviz.fr/Research/Graphcuisine>

5.2. Histomages

Participants: Fanny Chevalier, Pierre Dragicevic [correspondant], Christophe Hurter.

Histomages is an image editor based on a new interaction model that considers histogram views as spatial rearrangements of image pixels. Users can select pixels on image histograms as they would select image regions and directly manipulate them to adjust their colors. Histomages are affected by other image tools such as paintbrushes. We explored some possibilities offered by this interaction model, and discussed the four key principles behind it as well as their implications for the design of feature-rich software in general. <http://www.aviz.fr/histomages/>.

5.3. Gliimpse

Participants: Pierre Dragicevic [correspondant], Stéphane Huot, Fanny Chevalier.

Gliimpse is a quick preview technique that smoothly transitions between document markup code (HTML, LaTeX,...) and its visual rendering. This technique allows users to regularly check the code they are editing in-place, without leaving the text editor. This method can complement classical preview windows by offering rapid overviews of code-to-document mappings and leaving more screen real-estate. A proof-of-concept editor can be downloaded for free at <http://www.aviz.fr/gliimpse/>.

5.4. The Obvious Toolkit

Participants: Pierre-Luc Hémerly, Jean-Daniel Fekete [correspondant].

Information Visualization, Java, Toolkit

The Obvious Toolkit is a new Interactive Graphics Toolkit written in Java to facilitate the interoperability between Information Visualization toolkits and components (Fig. 4).

The Obvious Toolkit is an abstraction layer above visualization toolkits. Currently, it connects the most popular toolkits in Java: Prefuse, the InfoVis Toolkit, Improvise, JUNG, as well as other libraries such as the Java Database Communication Toolkit (JDBC) and two Machine-Learning toolkits: Weka and RapidMiner.

It is meant to provide an abstraction layer for information visualization application builders so that they can postpone their choice of a concrete toolkit to use. When faced with the choice, application builders can use one of the toolkits or connect all of them dynamically to Obvious. A paper on Obvious was presented at the IEEE Visual Analytics Science and Technology conference (VAST 2011) [61]. Obvious is available at <http://code.google.com/p/obvious>.

5.5. GeneaQuilts

Participants: Jean-Daniel Fekete [correspondant], Pierre Dragicevic, Anastasia Bezerianos, Julie Bae, Ben Watson.

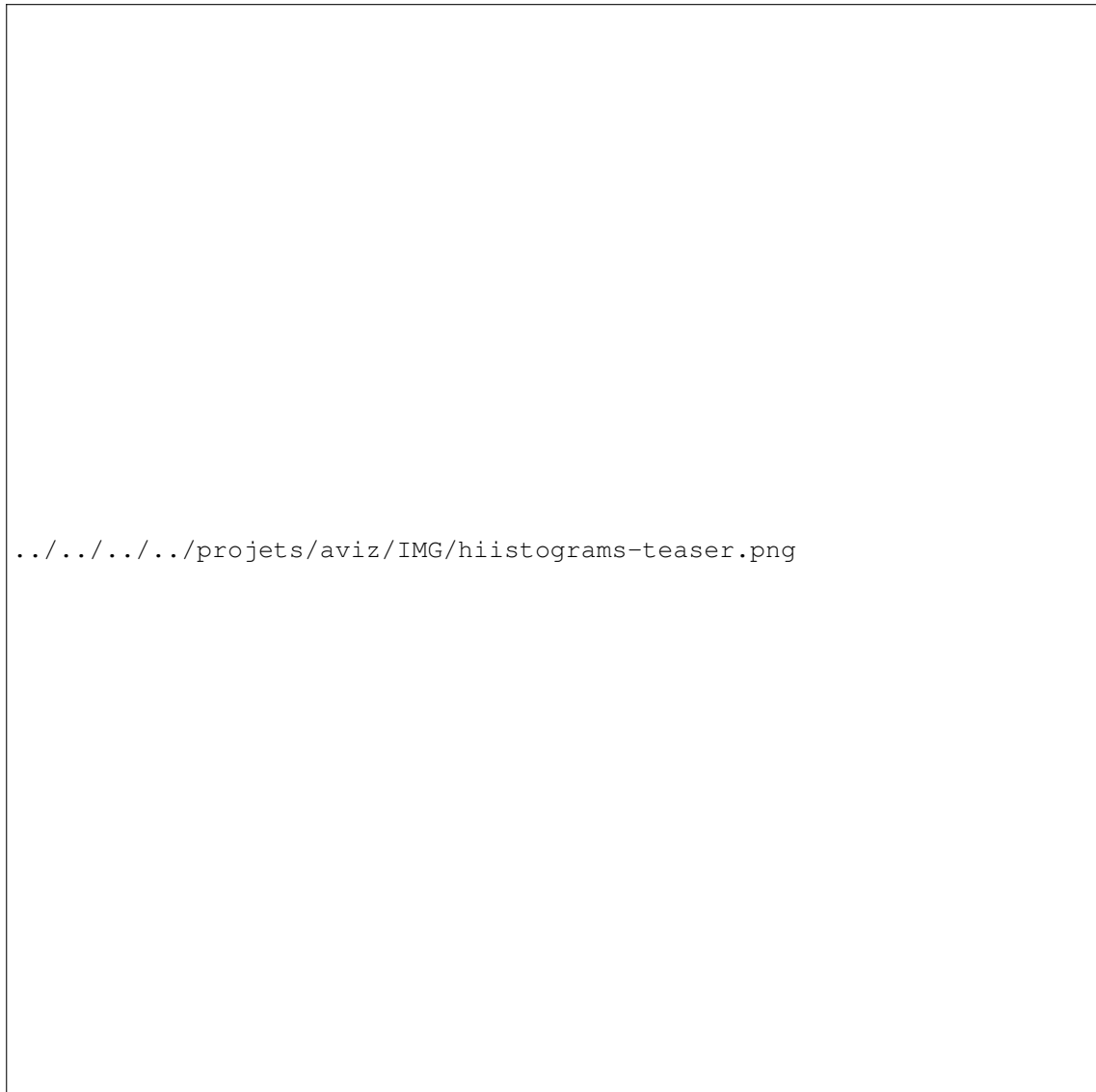


Figure 2. Example of sky enhancement with Histomages: (a) the image is duplicated and its pixels rearranged into a lightness histogram ; (b) bright pixels are selected with the rubber-band selection tool; (c) all pixels are rearranged into a hue histogram and yellow pixels are filtered out with the subtract selection brush (bottom). Missing pixels are added with the add selection brush on the image (top); (d) the sky is enhanced by resizing the selection on the saturation histogram.



Figure 3. Gliimpse: A detail of the animation between an article and its LaTeX source code.



Figure 4. The Obvious toolkit showing the same graph with a Prefuse and an IGTK rendering.

GeneaQuilts [2] is a new genealogy exploration software that allows genealogists and historians to visualize and navigate in large genealogies of up to several thousand individuals (Fig. 5). The visualization takes the form of a diagonally-filled matrix, where rows are individuals and columns are nuclear families. The GeneaQuilts system includes an overview, a timeline, search and filtering components, and a new interaction technique called Bring & Slide that allows fluid navigation in very large genealogies. The tool has been featured in several InfoVis and genealogy Websites and the website has been visited over 9000 times. It has been integrated in commercial and open-source implementations (4 to date). See also the web page <http://www.aviz.fr/geneaquilts/>.

5.6. Diffamation

Participants: Fanny Chevalier, Pierre Dragicevic [correspondant], Anastasia Bezerianos, Jean-Daniel Fekete.

Animation, Edit histories, Wikipedia, Revision Control

The Diffamation system [3] allows rapid exploration of revision histories such as Wikipedia or subversion repositories by combining text animated transitions with simple navigation and visualization tools. Diffamation can be used for example to get a quick overview of the entire history of a Wikipedia article or to see what has happened to one's contributions. Diffamation complements classical diff visualizations: once moments of interest have been identified, classical diff visualizations can come in useful to compare two given revisions in detail.

The Diffamation revision exploration system has been presented at the plenary session of the Ubuntu Developer Summit. It is available at <http://www.aviz.fr/diffamation/>.

5.7. The InfoVis Toolkit

Participant: Jean-Daniel Fekete [correspondant].

Information Visualization, Java, Toolkit



Figure 5. The genealogy of the Simpsons family (left) and of the Greek Pantheon (right), produced by the GeneaQuilts software.



Figure 6. Screenshot the Diffamation system during a transition: (a) the document view, (b) the overview scrollbar and (c) the timeline.

The InfoVis Toolkit [60] is an Interactive Graphics Toolkit written in Java to facilitate the development of Information Visualization applications and components.

The InfoVis Toolkit implements several visualization techniques, as well as interaction techniques related. It has been used for teaching the Information Visualization course (Masters level, Univ. of Paris-Sud) and is the basis for all AVIZ contracts. It is our main development platform for information visualization; most of our Information Visualization prototypes rely on it. It is available at <http://ivtk.sourceforge.net>.

In the forthcoming years, it will be superseded by extensions of the Obvious Toolkit (see section 5.4).

5.8. GraphDice

Participants: Jean-Daniel Fekete [correspondant], Pierre Dragicevic, Niklas Elmqvist, Anastasia Bezerianos.

GraphDice [1] is a visualization system for exploring multivariate networks (Fig. 7). GraphDice builds upon our previous system ScatterDice (best paper award at the IEEE InfoVis 2008 conference) [59]: it shows a scatter plot of 2 dimensions among the multiple ones available and provides a very simple paradigm of 3D rotation to change the visualized dimensions. The navigation is controlled by a scatter plot matrix that is used as a high-level overview of the dataset as well as a control panel to switch the dimensions.

While ScatterDice works on any tabular dataset (e. g., CSV file), the GraphDice system show networks using a node-link diagram representation as a scatter plot with links drawn between connected nodes. For more information, see the web page at <http://graphdice.gforge.inria.fr>.



Figure 7. Screenshot the GraphDice system.

HYBRID Project-Team

5. Software and Platforms

5.1. OpenViBE

Participants: Anatole Lécuyer [contact], Jozef Legény, Jussi Lindgren.

OpenViBE is a free and open-source software platform devoted to the design, test and use of Brain-Computer Interfaces (BCI). The platform consists of a set of software modules that can be integrated easily and efficiently to design BCI applications. The key features of OpenViBE software are its modularity, its high-performance, its portability, its multiple-users facilities and its connection with high-end/VR displays. The “designer” of the platform enables to build complete scenarios based on existing software modules using a dedicated graphical language and a simple Graphical User Interface (GUI). This software is available on the Inria Forge under the terms of the AGPL licence, and it was officially released in June 2009. Since then, the OpenViBE software has already been downloaded more than 12000 times, and it is used by numerous laboratories, projects, or individuals worldwide. The OpenViBE software is supported and improved in the frame of OpenViBE-NT project (section 8.2.9). More information, downloads, tutorials, videos, documentations are available on the [OpenViBE website](#).

5.2. GVT

Participants: Bruno Arnaldi, Valérie Gouranton [contact], Florian Novviale, Thomas Lopez.

The aim of GVT software (Generic Virtual Training) is to offer personalizable VR training sessions for industrial equipments. The main features of GVT software are the safeness offered by VR training (as opposed to trainind in risky real conditions), the optimization of the learning process, the creation of dedicated scenarios runnable on multiple hardware configurations: laptop or desktop computer, immersive room, distribution over network, etc. The current kernel of the GVT platform is divided into two main elements that rely on innovative models we have proposed: LORA (Language for Object-Relation Application) and STORM (Simulation and Training Object-Relation Model) models. With GVT behavioral engine, the objects of the virtual world expose behavioral capacities through the use of STORM. Then, the GVT scenario engine is used to determine the next steps of the procedure for a trainee, and its state evolves as the trainees achieve some actions, the scenario being written in LORA. As for today, a commercialized version of GVT, which includes a pedagogical engine developed at CERV laboratory, proposes training on individual procedures. In CORVETTE (section 8.2.1) and SIFORAS (section 8.2.6) projects, new features based on GVT are being designed, such as interactive, collaborative and physicalized actions, actors knowledge management, dialog using natural langage.

5.3. Collaviz

Participants: Thierry Duval [contact], Thi Thuong Huyen Nguyen.

The aim of Collaviz software (collaborative interactive visualization) is to allow to design, deploy and share collaborative virtual environments (CVE). Collaviz allows VR developpers to concentrate on the behavior of virtual objects that can be shared between users in a CVE. Indeed, Collaviz provides a software architecture that hides the network programmation details of the distribution and the synchronization of the content of the CVE, and that facilitates the coupling with the 3D graphics API used for rendering. Collaviz is written mainly in Java and is runnable on multiple hardware configurations: laptop or desktop computer, immersive room, mobile devices. The PAC-C3D software architecture of Collaviz makes it possible to use various 3D APIs for graphic rendering: Java3D, jReality, jMonkeyEngine, OpenSG, Unity3D (work in progress) and Havok Anarchy (work in progress), and also to use various physical engines such as jBullet and SOFA. The distribution over the network can be achieved using TCP or HTTP. An on-going collaboration with [Triskell team](#) intends to extend Collaviz using a Model Driven Engineering approach in order to provide high-level tools to generate a large part of java code of virtual objects.

IMAGINE Project-Team

4. Software and Platforms

4.1. MyCorporisFabrica

Participants: Ali-Hamadi Dicko, François Faure, Olivier Palombi.



Figure 1. My Corporis Fabrica is an anatomical knowledge database developed in our team.

My Corporis Fabrica (MyCF) is an anatomical knowledge database (see fig. 1). During 2011, we have added new anatomical entities and improved some parts of FMA (Foundational Model of Anatomy). The FMA's license is now under Creative Commons licenses (CC-by : Licensees may copy, distribute, display and perform the work and make derivative works based on it only if they give the author or licensor the credits in the manner specified by these). The license of MyCF is not yet defined. Our new contribution this year, is the creation of a brand new ontology about human functions. Based on the International Classification of Functioning, Disability and Health, also known as ICF, we have organized human functions through a tree of 4330 items. A original journal paper must be submitted soon. MyCF browser is now available on line: <http://www.mycorporisfabrica.org/>. The MyCf's generic programming framework can be used for other domains. The link with semantic and 3D models matches research activities of IMAGINE towards interactive digital creation media. Anatomy can be seen as a study case.

4.2. SOFA

Participants: François Faure, Ali Hamadi Dicko, Armelle Bauer, Olivier Carré, Matthieu Nesme, Romain Testylier, Moreno Trlin.



Figure 2. SOFA is an open source simulator for physically based modeling.

SOFA is a C++ library primarily targeted at medical simulation research. Based on an advanced software architecture, it allows to (1) create complex and evolving simulations by combining new algorithms with algorithms already included in SOFA; (2) modify most parameters of the simulation – deformable behavior, surface representation, solver, constraints, collision algorithm, etc. – by simply editing an XML file; (3) build complex models from simpler ones using a scene-graph description; (4) efficiently simulate the dynamics of interacting objects using abstract equation solvers; and (5) reuse and easily compare a variety of available methods.

SOFA is gaining momentum. A start-up based on SOFA, InSimo, has been created in Strasbourg by Inria people, and one of our former engineers, François Jourdes, has been hired. A SOFA-specific workshop was co-located with conference Vriphys'13 in Lille, with 50 attendants and the participation of several companies including CAE (a Canadian world leader in simulation), Haption, BASF, InSimo and others.

4.3. Expressive

Participants: Marie-Paule Cani, Amaury Jung, Mohamed-Galal Koraa, Maxime Quiblier, Cédric Zanni, Antoine Begault.



Figure 3. GUI and Example of implicit surface and modeled with the Expressive platform.

Expressive is a new C++ library developed to gather and share the models and algorithms developed within the ERC Expressive project. It enables us to make our latest research results on new creative tools; typically high level models together with intuitive, sketching or sculpting interfaces - soon available to the rest of the group and easily usable in our industrial partnerships. Its most developed part is Convol, a library dedicated implicit surfaces; and more particularly to the sub-classes of convolution surfaces and other integral surfaces along skeletons. Convol incorporates all the necessary material for constructive implicit modeling: skeleton-based convolution and SCALIS primitives, with closed form solution for the field values and gradient whenever possible; a variety of blending operators; and several methods for tessellating an implicit surface into a mesh, and for refining the later in highly curved regions. The creation of new geometry can be performed by direct manipulation of skeletal primitives or through sketch-based modeling.

IN-SITU Project-Team

5. Software and Platforms

5.1. WILD Platform

Participants: Michel Beaudouin-Lafon [correspondant], Olivier Chapuis, Stéphane Huot, Romain Primet, Amani Kooli, Monireh Sanaei, Gabriel Tezier, Jonathan Thorpe.

WILD (Wall-Size Interaction with Large Datasets) is InSitu's experimental ultra-high-resolution interactive platform for studying collaborative interaction and the visualization of very large datasets [2] (Figure 1). It features a wall-sized display with thirty-two 30" LCD screens, i.e. a 5m50 x 1m80 (18' x 6') wall displaying 20 480 x 6 400 = 131 million pixels, powered by a 16-computer cluster and two front-end computer. The platform also features a camera-based motion tracking system supporting interaction with the wall as well as within the surrounding space, a multitouch table and various mobile devices. WILD provides a unique experimental environment for interactive visualization and is part of the DIGISCOPE Equipment of Excellence. In addition to using WILD for our research, we have also developed software architectures and toolkits that enable developers to run applications on such multi-device, cluster-based systems.



Figure 1. The WILD platform (clockwise): wall display, motion tracking system, display cluster, interactive table.

5.2. jBricks

Participants: Stéphane Huot [correspondant], Mathieu Nancel, Romain Primet.

jBricks (Figure 2) is a Java toolkit that integrates a high-quality 2D graphics rendering engine based on ZVTM [46] and a versatile input configuration module (based on ICon [42] and FlowStates 5.4) into a coherent framework, enabling the exploratory prototyping of interaction techniques and rapid development of post-WIMP applications running on cluster-driven interactive visualization platforms such as wall-sized displays. The goal of this framework is to ease the development, testing and debugging of interactive visualization applications. It also offers an environment for the rapid prototyping of novel interaction techniques and their evaluation through controlled experiments.

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See E. Pietriga, S. Huot, M. Nancel, R. Primet, Rapid Development of User Interfaces on Cluster-Driven Wall Displays with jBricks, EICS '11: Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems, pages 185-190, June 2011
- OS/Middleware: Java (Linux, Mac OS X, Windows)
- Required library or software: several, managed through Maven
- Programming language: Java



Figure 2. jBricks applications running on the WILD platform (32 tiles for a total resolution of $20\,480 \times 6\,400$ pixels). (a) Zoomed-in visualization of the North-American part of the world-wide air traffic network (1 200 airports, 5 700 connections) overlaid on NASA's Blue Marble Next Generation images ($86\,400 \times 43\,200$ pixels) augmented with country borders ESRI shapefiles. (b) Panning and zooming in Spitzer's Infrared Milky Way ($396\,032 \times 12\,000$ pixels). (c) Controlled laboratory experiment for the evaluation of mid-air multi-scale navigation techniques.

5.3. The SwingStates Toolkit

Participants: Caroline Appert [correspondant], Michel Beaudouin-Lafon.

SwingStates [36] is a library that adds state machines and a graphical canvas to the Java Swing user interface toolkit. It was motivated by the lack of widely disseminated toolkits that support advanced interaction techniques and the observation that HCI research toolkits are little used outside the lab. By extending the popular Java Swing toolkit rather than starting from scratch, the goal is to facilitate the dissemination and adoption of SwingStates by practitioners.

SwingStates uses *state machines* to specify interaction. It provides programmers with a natural syntax to specify state machines and reduces the potential for an explosion of the number of states by allowing multiple state machines to work together or separately. SwingStates can be used to add new interaction techniques to existing Swing widgets, e.g. to select buttons and checkboxes by crossing rather than clicking. It can also be used with the SwingStates canvas (see below) and to control high-level dialogues.

SwingStates also provides a powerful *canvas widget*. The canvas can contain any Java2D shape, including geometric shapes, images, text strings and even Swing widgets. Shapes can be manipulated individually or collectively, through *tags*. An intensive use of polymorphism allows to apply almost any command to a tag: the command is then applied to all objects with this tag. Tags are also used in conjunction with state machines, to specify transitions that occur only on objects with a given tag. For example, pie menus can be implemented by creating a canvas in the overlay layer of any Swing application (Figure 3).



Figure 3. A numeric text field whose value can be set by a joystick-like interaction (left) and a semi-transparent menu to change the background color of Swing widgets (right)

SwingStates tightly integrates state machines, the Java language and the Swing toolkit to provide programmers with a natural and powerful extension to their natural programming environment. SwingStates is available at <http://swingstates.sf.net> under the GNU Lesser General Public License (LGPL).

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See C. Appert and M. Beaudouin-Lafon (2008) SwingStates: Adding State Machines to Java and the Swing Toolkit. *Software: Practice and Experience*, 38(11):1149 - 1182.
- OS/Middleware: Mac OS X, Linux, Windows
- Required library or software: Java virtual machine
- Programming language: Java

5.4. The FlowStates Toolkit

Participants: Caroline Appert [correspondant], Michel Beaudouin-Lafon, Stéphane Huot.

FlowStates [37], is a new toolkit to program advanced interaction techniques which require non standard input (e.g., two different mice that act independently, a joystick, a tablet, etc.). It is built on top of two existing toolkits: SwingStates [36] and ICon [42].



Figure 4. State machines and data flow in FlowStates

With FlowStates the developer can program interaction logic using state machines like SwingStates does but does not restrict the set of possible input channels to Java AWT standard input (a single couple <mouse, keyboard>). The state machines just have to define the virtual input events that are required to trigger their transitions so that FlowStates turns these machines into ICon devices which can be plugged to any physical input channels (Figure 4). An ICon device is a data flow building block that has input and output slots in order to be connected to other devices in the simple graphical environment provided by ICon. State machines can also send out events which appear as output slots in the data flow model.

With FlowStates we showed how two models for programming interaction (state machines and data flow) can be fully integrated to offer a huge power of expression. The explicit decision to not set strict limits between the roles of each model makes this hybrid approach highly flexible, the developer setting himself the limit between the two according to his needs and habits.

FlowStates is available at <http://www.lri.fr/~appert/FlowStates/>.

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See C. Appert, S. Huot, P. Dragicevic and M. Beaudouin-Lafon (2009) FlowStates: Prototypage d'applications interactives avec des flots de données et des machines à états. In Proceedings of IHM 2009. ACM, pages 119-128.
- OS/Middleware: Mac OS X, Linux, Windows
- Required library or software: ICon, Java virtual machine
- Programming language: Java

5.5. TouchStone

Participants: Caroline Appert [co-correspondant], Michel Beaudouin-Lafon, Wendy Mackay [co-correspondant].

TouchStone [8] is a platform for designing, running and analyzing the results of controlled experiments (Figure 5). While it focuses on experiments comparing interaction techniques, it can be used in a wide variety of contexts.



Figure 5. The architecture of the Touchstone platform

The *Touchstone Design platform* allows an experimenter to specify the factors, levels and measures in a controlled experiment, supports blocking and counterbalancing of trials and calculates how long it will take to run the experiment. Experimenters can compare the trade-offs between different experiment designs. The platform produces an XML file that serves as a protocol for the experiment and can be used as input to the Run platform.

The *Touchstone Run platform* provides a framework for implementing and running an experiment and collecting performance data. The flexible plug-in architecture supports various input devices and interaction techniques. The XML script from the Design platform can be run directly or edited to accommodate needs from specific experiments.

Log data from the Run platform can be analyzed by standard statistics tools such as JMP, R or Excel. In future, we hope to create a more elaborate *Touchstone Analysis platform* that will generate analysis scripts based on the output of the Design platform.

Members of InSitu use Touchstone for a variety of experiments and Students in the Research Masters (M2R Interaction) have been using it to design and implement experiments since 2011. Touchstone is available at <http://code.google.com/p/touchstone-platforms/> under a BSD License.

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See W. Mackay, C. Appert, M. Beaudouin-Lafon, O. Chapuis, Y. Du, JD. Fekete and Y. Guiard (2007) TouchStone: Exploratory Design of Experiments. In Proceedings of ACM CHI 2007 Conference on Human Factors and Computing Systems. ACM, pages 1425-1434.
- OS/Middleware: Mac OS X, Linux, Windows
- Required library or software: Java virtual machine
- Programming language: Java

5.6. Metisse

Participant: Olivier Chapuis [correspondant].

Metisse [40] is a window system that facilitates the design, implementation and evaluation of innovative window management techniques. The system is based on a compositing approach, making a clear distinction between the rendering and the interactive compositing processes. The Metisse server is a modified X server that supports both input and output redirection. The default compositor is a combination of a slightly modified version of FVWM, a standard window manager, with an interactive viewer application called *FvwmCompositor*.

FvwmCompositor uses OpenGL to display windows, which offers a rich graphics model well adapted to the exploration of new window management techniques. Texture mapping, for example, makes it possible to transform the window shapes in real-time (Figure 6, left). Alpha blending makes it easy to create translucent objects and shadows. Scaling, rotation and translation can also be used to position windows in 2D $\frac{1}{2}$ or 3D (Figure 6, middle and right). Input redirection makes it still possible to interact with applications no matter the visual transformations applied to the windows. It also makes it possible to adapt, reconfigure or re-combine existing graphical interfaces [48]. This year we used again Metisse to implement novel desktop interaction techniques [4].

- Web: <http://insitu.lri.fr/metisse/>
- ACM: H.5.2 [User Interfaces]: Windowing systems
- Software benefit: see [40], [48], [41], [44] and [4].
- License: GPL
- Type of human computer interaction: Graphique
- OS/Middleware: X Window et Mac OS X
- Required library or software: OpenGL via nucleo¹ and some usual C/C++ libraries
- Programming language: * C/C++

¹<http://interaction.lille.inria.fr/~rousseau/projects/nucleo/index.html>



Figure 6. Sample window management techniques implemented with Metisse: extended paper metaphor (left), interactive table configuration that allows to duplicate and rotate windows (middle) and zoomable 3D desktop (right).

5.7. The Substance Middleware

Participants: Michel Beaudouin-Lafon [correspondant], Clemens Klokmoose, Tony Gjerlufsen, James Eagan, Clement Pillias.

Substance is a middleware based on a novel programming paradigm called *data-oriented programming* and was designed to facilitate the development of multi-surface interactive applications [45]. Such applications are distributed by nature as they involve a varying number of display and interaction surfaces that are controlled by different computers. For example, our WILD room includes a 32-monitor display wall driven by 16 computers plus a front-end, a multi-touch table, various mobile devices such as iPodTouch and iPads, and the laptops that the users of the room may bring with them. We want to support seamless interaction techniques across these surfaces, such as the pick-and-drop technique pioneered by Rekimoto [47].

Data-oriented programming consists of attaching functionality to a tree data structure through *facets* attached to the individual nodes of the tree. Facets can be added and removed dynamically, and notified of changes in the tree. Substance supports two powerful ways to share nodes and facets: mounting, where access to the shared tree is managed through remotely, and replication, where the shared tree is replicated at each site and synchronized.

Substance has been used to create two full-scale applications (Figure 7): a generalized Canvas that can display and manage graphics, PDF files, image files and other content (through an extensible content manager) across surfaces spanning multiple displays and computers; SubstanceGrise, which uses multiple instances of the Anatomist/BrainVISA application to display coordinated 3D imagery of many brains in parallel on the WILD wall and control from a physical model of the brain.

Substance is available at <http://substance-env.sourceforge.net/> under a GNU GPL 3.0 licence.

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See T. Gjerlufsen, C. Klokmoose, J. Eagan, C. Pillias and M. Beaudouin-Lafon (2011) Shared Substance: Developing Flexible Multi-Surface Applications. In CHI '11: Proceedings of the 29th international conference on Human factors in computing systems. ACM, pages 3383-3392.
- OS/Middleware: Mac OS X, Linux
- Required library or software: several, managed by Python install
- Programming language: Python

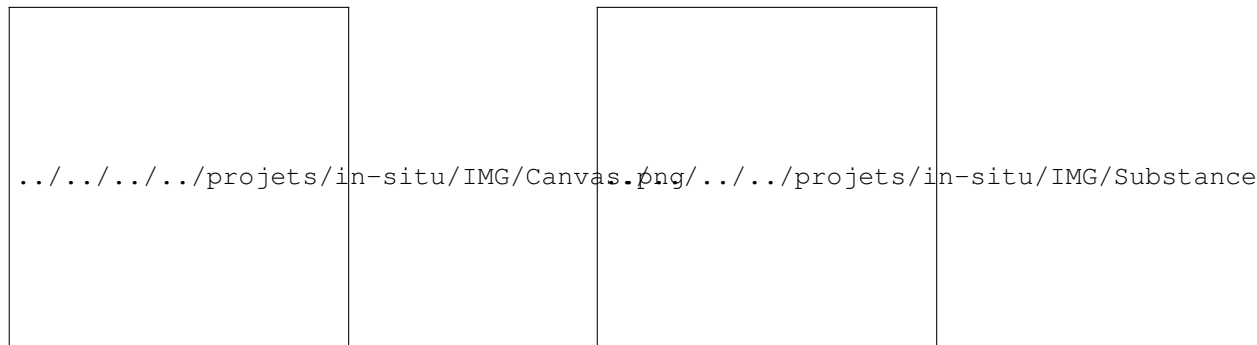


Figure 7. The Canvas (left) and SubstanceGrise (right) applications developed with Substance.
(©CNRS-Phototheque - Cyril FRESILLON for SubstanceGrise).

5.8. Scotty

Participants: Michel Beaudouin-Lafon [correspondant], James Eagan, Wendy Mackay.

The goal of Scotty is to support *malleable interfaces*, i.e. interfaces that can be modified at run-time in ways not anticipated by the designers [43]. Scotty is a toolkit that allows a programmer to extend an existing Mac OS X application without access to its source code. Scotty provides the following abstractions: hooks to alter the appearance of windows and widgets, event funnels to alter their behavior, glass sheets to overlay graphics and add new interaction methods, dynamic code loading and object proxies to redefine and extend existing objects. Scotty also provides a higher-level interface based on instrumental interaction [38]. Scotty currently runs on Mac OS X for applications written with the Cocoa user interface framework.



Figure 8. Using Scotty to teleport a window of a Mac OS X application onto an iPhone (left) and to create a toolglass in the Pages word processor (right).

Scotty has been used to create a number of extensions (Figure 8). *Scribbler* is a generic extension that uses glass sheets to allow handwritten annotations of any Cocoa window. *Teleportation* is another generic extension that can teleport and resize the content of any Cocoa window onto another computer, including an iPhone or iPad. The user can interact with the teleported content as if it was on the original computer. It was used to create a content provider for the Substance Canvas (see above), making it possible to display any application

running on a laptop onto the WILD wall display and/or table. When vector-based content is available, e.g., for text, Scotty provides smooth rescaling without the typical pixelation apparent when enlarging bitmap images. Finally *Stylesheet* is an extension to the Pages word processor that provides a semi-transparent toolglass for specifying the styles of paragraphs.

Scotty is available at <http://insitu.lri.fr/Projects/Scotty> under a GNU GPL 3.0 licence.

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See J. Eagan, W. Mackay and M. Beaudouin-Lafon (2011) Cracking the Cocoa Nut: User Interface Programming at Runtime. In UIST 2011: Proceedings of the 24th ACM Symposium on User Interface Software and Technology. ACM, pages 225-234.
- OS/Middleware: Mac OS X
- Required library or software: none
- Programming language: Objective-C, Python

MANAO Team

4. Software and Platforms

4.1. EIGEN

Participants: G. Guennebaud, D. Nuyts

Keywords : Linear algebra

Efficient numerical computation is central to many computer science domains. In particular, in computer graphics, space transformations and local regressions involve dense linear algebra, data interpolation and differential equations require sparse linear algebra, while more advanced problems involve non-linear optimization or spectral analysis. On the one hand, solutions such as MatLab are limited to prototyping. On the other hand, optimized libraries coming from the HPC (high performance computing) world are often tedious to use and more adapted for very large problems running on clusters. Moreover, all these solutions are very slow at handling very small but numerous problems which often arise in computer graphics, vision, or robotics. As a result, researchers of these domains used to waste a lot of time at either implementing their own half cooked solution, or dealing with dozens of complex to use libraries.

The objective of Eigen is to fill this gap by proposing an easy to use, efficient, and versatile C++ mathematical template library for linear algebra and related algorithms. In particular it provides fixed and dynamic size matrices and vectors, matrix decompositions (LU, LLT, LDLT, QR, eigenvalues, etc.), sparse matrices with iterative and direct solvers, some basic geometry features (transformations, quaternions, axis-angles, Euler angles, hyperplanes, lines, etc.), some non-linear solvers, automatic differentiations, etc. Thanks to expression templates, Eigen provides a very powerful and easy to use API. Explicit vectorization is performed for the SSE, AltiVec and ARM NEON instruction sets, with graceful fallback to non-vectorized code. Expression templates allow to perform global expression optimizations, and to remove unnecessary temporary objects.

Eigen is already a well established library with about 30k unique visitors of the website per month. Eigen is co-developed and maintained with a couple of other researchers and occasional contributors spread over the world. Its development started in 2008, and the last release is the 3.2 version in July 2013. Eigen has been supported by Inria through an ADT started in January 2012, and that ended in September 2013. This year, Eigen received the **“high-quality software in geometry processing award”** from the Symposium on Geometry Processing 2013 which was held in Genova, Pisa.

Facts:

- Web: <http://eigen.tuxfamily.org/>
- License: MPLv2

4.2. PatateLib

Participants: N. Mellado, G. Ciaudo, G. Guennebaud, P. Barla

Keywords : multi-scale analysis, material appearance, vector graphics, expressive rendering, 2D animation

Patate is a header only C++/CUDA library for graphics applications released under the MPL license.

It provides a collection of Computer Graphics techniques that incorporate the latest innovations from Inria research teams working in the field. It strives for efficiency and ease-of-use by focusing on low-level core operators and key algorithms, organised in modules that each tackle a specific set of issues. The central goal of the library is to drastically reduce the time and efforts required to turn a research paper into a ready-to-use solution, for both commercial and academic purposes.

Each module is initially developed by a few persons, usually those who have authored the corresponding research papers. An engineer, Gautier Ciaudo, has been recruited via the ADT program to perform unit tests, bug tracking, and make examples. Our first module provides efficient methods for the fitting and analysis of point-clouds in arbitrary dimensions. It may be used for varied purposes such as curvature computation, surface reconstruction, scale-space analysis, image processing, and sketch vectorization. More modules will be developed in 2014.

Facts:

- Web: <http://patate.gforge.inria.fr/html/index.html>
- License: MPLv2

4.3. PFSTools

Participant: I. Ihrke

Keywords : high dynamic range image processing, merging, calibration and tone-mapping

The `pfstools` package is a set of command line programs for reading, writing, manipulating and viewing high-dynamic range (HDR) images and video frames. All programs in the package exchange data using a simple generic high dynamic range image format, `pfs`, and they use unix pipes to pass data between programs and to construct complex image processing operations.

`pfstools` come with a library for reading and writing `pfs` files. The library can be used for writing custom applications that can integrate with the existing `pfstools` programs. It offer also a good integration with high-level mathematical programming languages, such as MATLAB or GNU Octave. `pfstools` can be used as the extension of MATLAB or Octave for reading and writing HDR images or simply to store effectively large matrices. The `pfstools` package is an attempt to integrate the existing high dynamic range image formats by providing a simple data format that can be used to exchange data between applications. It is accompanied by the `pfscalibration` and `pfstmo` packages.

The `pfscalibration` package provides an algorithm for the photometric calibration of cameras and for the recovery of high dynamic range (HDR) images from the set of low dynamic range (LDR) exposures. Maintenance of the `pfscalibration` package is done by Ivo Ihrke since January 2011. A major update to make the software compatible with current digital SLR cameras and their raw file formats, especially for measurement purposes, has been performed. A new set of MATLAB scripts has been developed for improved calibration performance. It is intended to merge these new procedures into the existing software.

The `pfstmo` package contains the implementation of seven state-of-the-art tone mapping operators suitable for convenient processing of both static images and animations.

The software received wider interest of the Open Source community and third party contributors prepared installation packages which are included in several Linux distributions including Debian, Fedora and Suse.

Facts:

- Web: <http://pfstools.sourceforge.net/>
- License: GPL

MAVERICK Project-Team

5. Software and Platforms

5.1. Introduction

Maverick insists on sharing the software that is developed for internal use. These are all listed in a dedicated section on the web site <http://artis.imag.fr/Software>.

5.2. Gratin

Participant: Romain Vergne [contact].

Gratin is a node-based compositing software for creating, manipulating and animating 2D and 3D data. It uses an internal direct acyclic multi-graph and provides an intuitive user interface that allows to quickly design complex prototypes. Gratin has several properties that make it useful for researchers and students. (1) it works in real-time: everything is executed on the GPU, using OpenGL, GLSL and/or Cuda. (2) it is easily programmable: users can directly write GLSL scripts inside the interface, or create new C++ plugins that will be loaded as new nodes in the software. (3) all the parameters can be animated using keyframe curves to generate videos and demos. (4) the system allows to easily exchange nodes, group of nodes or full pipelines between people. In a research context, Gratin aims at facilitating the creation of prototypes, testing ideas and exchanging data. For students, Gratin can be used to show real-time demos/videos, or help learning how to program with the GPU. Gratin has already been used for creating new computer graphics tools but also for designing perceptual experiments. Most of the work published by R. Vergne was done with Gratin.

5.3. PlantRad

Participant: Cyril Soler [contact].

PlantRad is a software program for computing solutions to the equation of light equilibrium in a complex scene including vegetation. The technology used is hierarchical radiosity with clustering and instantiation. Thanks to the latter, PlantRad is capable of treating scenes with a very high geometric complexity (up to millions of polygons) such as plants or any kind of vegetation scene where a high degree of approximate self-similarity permits a significant gain in memory requirements. Its main domains of applications are urban simulation, remote sensing simulation (See the collaboration with Noveltis, Toulouse) and plant growth simulation, as previously demonstrated during our collaboration with the LIAMA, Beijing.

5.4. High Quality Renderer

Participant: Cyril Soler [contact].

In the context of the European project RealReflect, the Maverick team has developed the HQR software based on the photon mapping method which is capable of solving the light balance equation and of giving a high quality solution. Through a graphical user interface, it reads X3D scenes using the X3DToolkit package developed at Maverick, it allows the user to tune several parameters, computes photon maps, and reconstructs information to obtain a high quality solution. HQR also accepts plugins which considerably eases the development of new algorithms for global illumination, those benefiting from the existing algorithms for handling materials, geometry and light sources. HQR is freely available for download at <http://artis.imag.fr/~Cyril.Soler/HQR>.

5.5. MobiNet

Participants: Fabrice Neyret [contact], Joëlle Thollot.



Figure 2. Gratin interface.

The MobiNet software allows for the creation of simple applications such as video games, virtual physics experiments or pedagogical math illustrations. It relies on an intuitive graphical interface and language which allows the user to program a set of mobile objects (possibly through a network). It is available in public domain at <http://mobinet.inrialpes.fr> for Linux, Windows and MacOS, and originated in a collaboration with the EVASION project-team.

The main aim of MobiNet is to allow young students at high school level with no programming skills to experiment, with the notions they learn in math and physics, by modeling and simulating simple practical problems, and even simple video games. This platform has been massively used during the Grenoble INP "engineer weeks" since 2002: 150 senior high school pupils per year, doing a 3 hour practice. This work is partly funded by Grenoble INP. Various contacts are currently developed in the educational world. Besides "engineer weeks", several groups of "monitors" PhD students conducts experimentations based on MobiNet with a high school class in the frame of the courses. Moreover, presentation in workshops and institutes are done, and a web site repository is maintained. A web version is currently under preliminary development.

5.6. Freestyle

Freestyle is a software for Non-Photorealistic Line Drawing rendering from 3D scenes (Figure 3). It is designed as a programmable interface to allow maximum control over the style of the final drawing: the user "programs" how the silhouettes and other feature lines from the 3D model should be turned into stylized strokes using a set of programmable operators dedicated to style description. This programmable approach, inspired by the shading languages available in photorealistic renderers such as Pixar's RenderMan, overcomes the limitations of integrated software with access to a limited number of parameters and permits the design of an infinite variety of rich and complex styles. The system currently focuses on pure line drawing as a first step. The style description language is Python augmented with our set of operators. Freestyle was developed in the framework of a research project dedicated to the study of stylized line drawing rendering from 3D scenes. This research has led to two publications [25], [26].

In 2008, Freestyle get a new life, completely outside Maverick or Inria: it was the basis of one of the 6 *Google Summer of Code* projects awarded to the *Blender Foundation*¹! The goal of the project was to integrate Freestyle to the well known free 3D modeler *Blender*, as its standard NPR line-drawing renderer. Maxime Curioni (under the mentoring of Jean-Luc Peurière from the *Blender Foundation*), is currently making the integration. First beta versions are publicly available, and tested by enthusiasts around the web.

5.7. Diffusion Curves

Participant: Joëlle Thollot [contact].

We provide an implementation of the vector drawing tool described in our Diffusion Curves Siggraph paper [2] (Figure 4). This prototype is composed of the Windows binary, along with the required shader programs (ie. in source code). The software is available for download at <http://artis.imag.fr/Publications/2008/OBWBTS08> for free, for non-commercial research purposes.

5.8. VRender: vector figures

Participant: Cyril Soler [contact].

The VRender library is a simple tool to render the content of an OpenGL window to a vectorial device such as Postscript, XFig, and soon SVG. The main usage of such a library is to make clean vectorial drawings for publications, books, etc.

In practice, VRender replaces the z-buffer based hidden surface removal of OpenGL by sorting the geometric primitives so that they can be rendered in a back-to-front order, possibly cutting them into pieces to solve cycles.

¹<http://www.blender.org/>



Figure 3. Stylized plane using Freestyle.



Figure 4. Diffusion curves freely downloadable demo.

VRender is also responsible for the vectorial snapshot feature of the QGLViewer library. VRender is released under the LGPL licence and is freely available for download at <http://artis.imag.fr/Software/VRender>.

5.9. ProLand

Participants: Fabrice Neyret [contact], Eric Bruneton.

Now available at <http://proland.inrialpes.fr/> in double licencing GPL/commercial.

Proland (for procedural landscape) is a software platform originally developed at the Evasion team-project by Eric Bruneton, and currently funded by the ANR-JCJC SimOne. The goal of this platform is the real-time quality rendering and editing of large landscapes. All features can work with planet-sized terrains, for all viewpoints from ground to space. Most of the work published by Eric Bruneton and Fabrice Neyret has been done within Proland, and a large part has been integrated in the main branch. Several licences have been transferred to companies. Eric Bruneton was hired by Google-Zürich in september 2011, but will be able to keep some participation in the project.

5.10. Giga Voxels

Participants: Fabrice Neyret [contact], Goswami Prashant, Sinoir Jérémy, Cyril Crassin, Pascal Guehl, Eric Heitz.

Soon available at <http://gigavoxels.inrialpes.fr/index.htm> in double licencing GPL/commercial.

Gigavoxel is a software platform initiated from the PhD work of Cyril Crassin, and currently funded by the ANR CONTINT RTIGE (Figure 5). The goal of this platform is the real-time quality rendering of very large and very detailed scenes which couldn't fit memory. Performances permit showing details over deep zooms and walk through very crowded scenes (which are rigid, for the moment). The principle is to represent data on the GPU as a Sparse Voxel Octree which multiscale voxels bricks are produced on demand only when necessary and only at the required resolution, and kept in a LRU cache. User defined producer lays across CPU and GPU and can load, transform, or procedurally create the data. Another user defined function is called to shade each voxel according to the user-defined voxel content, so that it is user choice to distribute the appearance-making at creation (for faster rendering) or on the fly (for storageless thin procedural details). The efficient rendering is done using a GPU differential cone-tracing using the scale corresponding to the 3D-MIPmapping LOD, allowing quality rendering with one single ray per pixel. Data is produced in case of cache miss, and thus only whenever visible (accounting for view frustum and occlusion). Soft-shadows and depth-of-field is easily obtained using larger cones, and are indeed cheaper than unblurred rendering. Beside the representation, data management and base rendering algorithm themself, we also worked on realtime light transport, and on quality prefiltering of complex data. Ongoing researches are addressing animation. GigaVoxels is currently used for the quality real-time exploration of the detailed galaxy in ANR RTIGE. This work led to several publications and several licences have been sold to companies.



Figure 5. GigaVoxels freely downloadable demo.

MIMETIC Project-Team

5. Software and Platforms

5.1. HPTS++: Hierarchical Parallel Transition System ++

Participant: Fabrice Lamarche [contact].

APP deposit number: IDDN.FR.001.290017.000.S.P.2003.000.10400

HPTS++ is a platform independent toolkit to describe and handle the execution of multi-agent systems. It provides a specific object oriented language encapsulating C++ code for interfacing facilities and a runtime kernel providing automatic synchronization and adaptation facilities.

The language provides functionalities to describe state machines (states and transitions) and to inform them with user specific C++ code to call at a given point during execution. This language is object oriented and supports concepts such as polymorphism and inheritance (state machines and user defined C++ classes). The compilation phase translates a state machine in a C++ class that can be compiled separately and linked through static or dynamic libraries. The runtime kernel includes a scheduler that handles parallel state machines execution and that provides synchronization facilities such as mutual exclusion on resources, dead lock avoidance, notions of priorities and execution adaptation in accordance with resources availability.

HPTS++ also provides a task model. Thanks to this model, the user can describe primitive behaviors through atomic tasks and combine them with operators (e.g. sequence, parallelism, loops, alternatives). These operators are fully dynamic. Hence they can be used at runtime to rapidly create complex behaviors.

5.2. MKM: Manageable Kinematic Motions

Participants: Richard Kulpa [contact], Franck Multon.

APP deposit numbers: IDDN.FR.000.290016.001.S.A.2003.000.41200, IDDN.FR.001.290016.001.S.A.2003.000.41200 et IDDN.FR.002.290016.001.S.A.2003.000.41200

We have developed a framework for animating human-like figures in real-time, based on captured motions. This work was carried-out in collaboration with the M2S Laboratory (Mouvement, Sport, Santé) of the University Rennes 2.

In this software, we propose a morphology-independent representation of the motion that is based on a simplified skeleton which normalizes the global postural informations. This formalism is not linked to morphology and allows very fast motion retargetting and adaptation to geometric constraints that can change in real-time. This approach dramatically reduces the post production time and allows the animators to handle a general motion library instead of one library per avatar.

The framework provides an animation library which uses the motions either obtained from our off-line tool (that transforms standard formats into our morphology-independent representation) or parameterized models in order to create complete animation in real-time. Several models are proposed such as grasping, orientation of the head toward a target. We have also included a new locomotion model that allows to control the character directly using a motion database.

In order to create realistic and smooth animations, MKM uses motion synchronization, blending and adaptation to skeletons and to external constraints. All those processes are performed in real-time in an environment that can change at any time, unpredictably.

All these features have been used to anticipate and control the placement of footprints depending on high level parameters. This link between control and behavior levels will be used for reactive navigation in order to have realistic motion adaptations as well as to deal with constrained environments.

5.3. TopoPlan: Topological Planner and Behaviour Library

Participant: Fabrice Lamarche [contact].

APP deposit numbers: FR.001.480016.00.S.P.2008.000.41200

TopoPlan (Topological Planner) is a toolkit dedicated to the analysis of a 3D environment geometry in order to generate suitable data structures for path finding and navigation. This toolkit provides a two step process: an off-line computation of spatial representation and a library providing on-line processes dedicated to path planning, environmental requests...

TopoPlan is based on an exact 3D spatial subdivision that accurately identifies floor and ceiling constraints for each point of the environment. Thanks to this spatial subdivision and some humanoid characteristics, an environment topology is computed. This topology accurately identifies navigable zones by connecting 3D cells of the spatial subdivision. Based on this topology several maps representing the environment are extracted. Those maps identify obstacle and step borders as well as bottlenecks. TopoPlan also provides a runtime library enabling the on-line exploitation of the spatial representation. This library provides several algorithms including roadmap-based path-planning, trajectory optimization, footprint generation, reactive navigation and spatial requests through customizable spatial selectors.

TopoPlan behavior is a library built on top of TopoPlan and MKM providing several behaviors described thanks to the HPTS++ task model. Its goal is to provide a high level interface handling navigation and posture adaptation within TopoPlan environments. Provided behaviors include:

- A behavior handling fully planned navigation toward an arbitrary destination. This behavior precisely handles footprint generation within constrained environments such as stairs for instance.
- A behavior controlling an MKM humanoid to follow a trajectory specified by the user.
- A behavior controlling MKM to follow a list of footprints given by the user.
- A behavior adapting the humanoid posture to avoid collision with ceiling. This behavior runs in parallel of all other behaviors and adapts humanoid motion when needed without any user intervention.
- A behavior handling reactive navigation of virtual humans. This behavior plan a path to a given target and follows the path while avoiding collisions with other navigating entities.

Those behaviors have been built using the HPTS++ task model. Thus, they can be easily combined together or with other described behaviors through task operators.

MINT Project-Team

5. Software and Platforms

5.1. LibGINA

Participant: Laurent Grisoni [correspondant].

This library has been developed within the context of the ADT GINA, for one of the installation that have been made in collaboration with Le Fresnoy national studio (Damassama, Léonore Mercier). This library is currently being posted as APP, and has been used by Idées-3com small company, in the context of our join I-lab program. This library allows for use of gesture for command, and is able to handle strong variability into recognized patterns.

Current version: version 1.0

Software characterization: A-2 SO-3 SM-2-up EM-3 SDL-3 OC-DA4-CD4-MS2-TPM4

5.2. 3D interaction using mobile phone

Participants: Samuel Degrande [correspondant], Laurent Grisoni.

This work has been achieved in the context of the Idées-3com I-lab. In this context a module, that allows to use any android based smartphone to control an Explorer module for navigation and interaction with VRML-based content. This module was used as a basis by Idées-3com in their commercial product this year.

Current version: version 1.0

Software characterization: A-2 SO-3 SM-2-up EM-2-up SDL-3 OC-DA4-CD4-MS2-TPM4

5.3. tIO (tactile input & output)

Participants: Marc-Antoine Dupré, Nicolas Roussel [correspondant], Takashi Miyaki.

tIO is a library designed to facilitate the implementation of doubly tactile interaction techniques (tactile input coupled with tactile feedback) based on the STIMTAC technology. Supporting all current STIMTAC prototypes, it makes it easy to move the system pointer of the host computer according to motions detected on them and adapt their vibration amplitude based on the color of the pointed pixel or the nature of the pointed object. The library includes a set of Qt demo applications that illustrate these two different approaches and makes it easy to “augment” existing Qt applications with tactile feedback. It also makes it possible to supplement or substitute tactile feedback with basic auditory feedback synthesized using `portaudio` (friction level is linearly mapped to the frequency of a sine wave). This not only facilitates the development and documentation of tactile-enhanced applications but also makes it easier to demonstrate them to a large audience.

Software characterization: A2, SO3-up, SM-2, EM2, SDL1.

5.4. libpointing

Participants: Géry Casiez [correspondant], Damien Marchal, Nicolas Roussel.

Libpointing is a software toolkit that provides direct access to HID pointing devices and supports the design and evaluation of pointing transfer functions [2]. The toolkit provides resolution and frequency information for the available pointing and display devices and makes it easy to choose between them at run-time through the use of URIs. It allows to bypass the system’s transfer functions to receive raw asynchronous events from one or more pointing devices. It replicates as faithfully as possible the transfer functions used by Microsoft Windows, Apple OS X and Xorg (the X.Org Foundation server). Running on these three platforms, it makes it possible to compare the replicated functions to the genuine ones as well as custom ones. The toolkit is written in C++ with Python and Java bindings available. It is publicly available under the GPLv2 license.

Web site: <http://libpointing.org/>

Software characterization: A3, SO3, SM-2, EM2, SDL4

5.5. Platform PIRVI

MINT is associated to the CPER-CIA (2007-2013), and participates to the PIRVI platform (Framework for Computer Human Animation, Virtual Reality and Images, handled by F. Aubert, co-animated by F. Aubert and D. Marchal), which aims at promoting research achieved by participant research teams (6 research teams, among which MINT), as well as encouraging collaborations with regional economical tissue on the knowledge fields covered within the associated research teams. The PIRVI allows these research teams to share a Virtual-Reality Room and various mid-size research equipments : multitouch tables, cameras (depth, infrared, ...), interactive devices (force-feedback, multitouch, smartphones...), a configurable multitouch wall. This dissemination activity has been supported with a regional contract 500 Keuros.

POTIOC Team

5. Software and Platforms

5.1. OpenViBE

Participants: Fabien Lotte [local correspondent], Alison Cellard [engineer].

As part of our research work on BCI, we contribute to the development of the OpenViBE² software, which is an open source platform dedicated to the design, evaluation and use of BCI for real and virtual applications. OpenViBE development is led by Inria, and Potioc is one of the Inria team contributing to its evolution. Moreover, Potioc is involved in the Inria ADT (Technological Development Action) OpenViBE-NT that is dedicated to the development of OpenViBE together with 3 other Inria teams (Hybrid, Athena, Neurosys).

5.2. Drile

Participant: Florent Berthaut.

As part of the research on Virtual Reality for Musical Performance, notably the Drile system, various software pieces are being developed and made available to the community. These software pieces are the following:

- Pure-Data external to access data from the Virtual Reality Peripheral Network : <https://github.com/scrime/vrpd>
- Drile: <http://hitmuri.net/index.php/Research/Drile>

5.3. 3DCityTestbed

Participants: Jacek Jankowski, Thomas Hulin.

As part of the research on the "Villes transparentes" project in collaboration with Mappy (Solocal group) and Vectuel - VirtuelCity, we develop a software platform dedicated to the design and evaluation of innovative interaction techniques for the navigation in urban 3D environments. This code is not publicly available yet.

5.4. PapARt

Participant: Jérémy Laviolle [Main developer].

As part of his thesis work, Jérémy Laviolle has developed a software suite for PapARt : Paper Augmented Reality Toolkit. This suite enables the calibration of depth cameras such as the Kinect with a planar surface and with a videoprojection. It also enables the detection of finger touch on the planar surface. This system can be extended to 3D objects. Many external devices are compatible, such as pen tablets (Wacom) and LEAP Motion. The code runs on Linux, Mac OS and Windows, and is made for Processing³. This code is not publicly available yet.

²<http://openvibe.inria.fr>

³<http://www.processing.org>

REVES Project-Team

4. Software and Platforms

4.1. Multi-View Image-Based Rendering and Relighting Suite

Participants: Clement Riant, Sylvain Duchêne, Pierre-Yves Laffont, Adrien Bousseau, George Drettakis.

We have designed and implemented a set of libraries for handling multi-view image-based rendering and relighting algorithms. These constitute the basis for the software developed for the EU projects VERVE and CR-PLAY.

4.1.1. RID: Rich Intrinsic-image Decomposer

We developed a software platform to perform rich intrinsic decomposition methods from photographs of outdoor scenes, as described in [13]. It includes main scripts and functions in Matlab for treatment of the input data, interfaces to software for multi-view reconstruction (Bundler, PMVS) and meshing from point clouds (method developed by Julie Digne, a postdoc in the GEOMETRICA project team). We then interface software for image matting using the Matting Laplacian, and User-Assisted Intrinsic Images. The system also includes an interface with Adobe Photoshop, for visualization and demonstration of our results in end-user image editing software. The method performs the computation of sun, sky and indirect lighting received at 3D points of an automatically reconstructed scene, using a modified version of the PBRT stochastic raytracer. Finally, there is a scene calibration module and an OpenGL viewer.

4.1.2. ROSSE: Relighting Outdoor Scenes with Shadow Editing

This software package includes a set of modules for processing point clouds and meshes produced by automatic multi-view stereo computer vision solutions. It includes all file management, point cloud and mesh handling, as well as ray-tracing using the Intel Embree ray tracer to compute illumination properties on the mesh. An interactive viewer is also included. A new intrinsic image approach is included as well as a module for relighting and shadow movement, based on an image-driven approach to moving cast shadows.

4.1.3. SWARPI: Superpixel Warp for Image-based rendering

Depth Synthesis and Warped-Based Superpixel Image-Based Rendering. This software package is the implementation of the publication [12]. The main software consists of two components: the depth synthesis step and the image-based rendering step:

a) The depth synthesis step is a Matlab package that reads 3D points coming from an automated 3D reconstruction pipeline, together with images and calibrated cameras, and produces the superpixel decomposition and the depth synthesis algorithm. The current version uses the open source packages bundler and PMVS (GPL v3 license), but other 3D reconstruction approaches could be used instead. b) The rendering step is in C++, and takes the result of the first step as input to allow interactive navigation. The code uses multi-pass deferred shading with geometry shaders (OpenGL 4.0 or above) to perform the rendering.

In addition to the implementation of [12], we have developed a Matlab interface for manual depth correction ("depth painting") An APP (Agency for the Protection of Programs) registration of this software is pending.

4.2. APF: state-of-the-art 3D audio library

Participants: Adrien David, George Drettakis.

This work was performed in collaboration with Jean-Christophe Lombardo of the DREAM group (i.e., the research support development group of our Inria center). REVES has several audio research publications over the last 10 years, which correspond to a class of functionalities such as clustering, masking, progressive processing etc.. The first component is the masking or culling algorithm, which aims at removing all the inaudible audio sources from a virtual scene based on perceptual metrics. The second component, called clustering, aims at grouping audio sources that are spatially close to each other and premix them to a representative cluster source, so that all spatialization related processing can be applied only on the representative premixed source [9]. Other audio topics were also considered and developed, like progressive and scalable frequency domain mixing, sound propagation, scalable reverberation, modal sound synthesis and contact sounds generation [1].

In order to maintain all the knowledge in the group and re-use these technologies in the Immersive Space, a previous young engineer (David Grelaud) wrote a fully documented audio library (APF) which gathers about 10 audio publications and 1 US patent. APF is a cross-platform, object oriented C++ API available on GForge. All the code has been re-implemented and a completely new software architecture resulted in a twofold increase in the speed of our algorithms. APF runs in the Immersive Space and uses the tracking system to spatialize virtual audio sources around the listener. It can also exploit personal Head Related Transfer Functions (HRTF).

We have implemented a network communications layer to create an audio rendering server on a separate machine, and the library is fully integrated into the osgVR platform.

APF has been critical in establishing collaborations in the context of various grant proposals (EU and national).

4.3. GaborNoise Software

Participants: Ares Lagae, George Drettakis.

We proposed a new procedural noise function last year, Gabor noise [6]. In the context of this project, we have developed a software package, which includes a CPU reference implementation of the 2D noise, and a complete GPU implementation of the 2D noise, surface noise, and 3D noise. This software package has been filed for APP protection and is in the process of being transferred to industrial partners.

This work is a collaboration with Sylvain Lefebvre, former member of the team, now in the ALICE project-team, Inria Nancy - Grand Est.

4.4. Gabor Noise By Example

Participant: George Drettakis.

In collaboration with B. Galerne, S. Lefebvre and A. Lagae (KU Leuven) we have released to code for the 2012 SIGGRAPH paper Gabor Noise By Example (see <http://www-sop.inria.fr/reves/Basilic/2012/GLLD12/>). This includes a matlab code for the analysis and C++/cuda code for the synthesis.

TITANE Team

5. Software and Platforms

5.1. CGAL, the Computational Geometry Algorithms Library

Participants: Pierre Alliez, Clement Jamin, Florent Lafarge, Sven Oesau, David Bommes.

CGAL is a C++ library of geometric algorithms and data structures. Our team is involved in several on-going implementations: parallelization of mesh generation and triangulations, shape detection in unstructured point sets, geodesic distances on surface meshes and barycentric coordinates (in collaboration with Dmitry Anisimov). Pierre Alliez is a member of the CGAL Editorial Board.

5.2. APP deposits

5.2.1. MeshMantics

Participants: Yannick Verdie, Florent Lafarge, Pierre Alliez.

MeshMantics is a software for segmenting 2-manifold surface meshes in an urban context. Four classes of interest are considered: ground, vegetation, roof and facades.

ALPAGE Project-Team

5. Software and Platforms

5.1. Syntax

Participants: Pierre Boullier [correspondant], Benoît Sagot.

See also the web page <http://syntax.gforge.inria.fr/>.

The (currently beta) version 6.0 of the SYNTAX system (freely available on Inria GForge) includes various deterministic and non-deterministic CFG parser generators. It includes in particular an efficient implementation of the Earley algorithm, with many original optimizations, that is used in several of Alpage's NLP tools, including the pre-processing chain SXPipe and the LFG deep parser SXLFG. This implementation of the Earley algorithm has been recently extended to handle probabilistic CFG (PCFG), by taking into account probabilities both during parsing (beam) and after parsing (n -best computation). SYNTAX 6.0 also includes parsers for various contextual formalisms, including a parser for Range Concatenation Grammars (RCG) that can be used among others for TAG and MC-TAG parsing.

Direct NLP users of SYNTAX for NLP, outside Alpage, include Alexis Nasr (Marseilles) and other members of the (now closed) SEQUOIA ANR project, Owen Rambow and co-workers at Columbia University (New York), as well as (indirectly) all SXPipe and/or SXLFG users. The project-team VASY (Inria Rhône-Alpes) is one of SYNTAX' user for non-NLP applications.

5.2. DyALog

Participant: Éric Villemonte de La Clergerie [maintainer].

DYALOG on Inria GForge: <http://dyalog.gforge.inria.fr/>

DYALOG provides an environment to compile and execute grammars and logic programs. It is essentially based on the notion of tabulation, i.e. of sharing computations by tabulating traces of them. DYALOG is mainly used to build parsers for Natural Language Processing (NLP). It may nevertheless be used as a replacement for traditional PROLOG systems in the context of highly ambiguous applications where sub-computations can be shared.

The current release **1.13.0** of DYALOG is freely available by FTP under an open source license and runs on Linux platforms for x86 and architectures and on Mac OS intel (both 32 and 64bits architectures).

The current release handles logic programs, DCGs (*Definite Clause Grammars*), FTAGs (*Feature Tree Adjoining Grammars*), FTIGs (*Feature Tree Insertion Grammars*) and XRCGs (*Range Concatenation Grammars* with logic arguments). Several extensions have been added to most of these formalisms such as intersection, Kleene star, and interleave operators. Typed Feature Structures (TFS) as well as finite domains may be used for writing more compact and declarative grammars [101].

C libraries can be used from within DYALOG to import APIs (mysql, libxml, sqlite, ...).

DYALOG is largely used within ALPAGE to build parsers but also derivative softwares, such as a compiler of Meta-Grammars (cf. 5.3). It has also been used for building FRMG, a parser from a large coverage French TIG/TAG grammar derived from a Meta-Grammar. This parser has been used for the Parsing Evaluation campaign EASy, the two Passage campaigns (Dec. 2007 and Nov. 2009), cf. [99], [100], and very large amount of data (700 millions of words) in the SCRIBO project. New results concerning FRMG are described in 6.11 .

A new statistical dependency parser, based on a shift-reduce algorithm, was also developed in 2013 within the DYALOG system (see 6.12).

DYALOG and other companion modules are available on Inria GForge.

5.3. Tools and resources for Meta-Grammars

Participant: Éric Villemonte de La Clergerie [maintainer].

mgcomp, *MGTOOLS*, and *FRMG* on Inria GForge: <http://mgkit.gforge.inria.fr/>

DYALOG (cf. 5.2) has been used to implement *mgcomp*, Meta-Grammar compiler. Starting from an XML representation of a MG, *mgcomp* produces an XML representation of its TAG expansion.

The current version **1.5.0** is freely available by FTP under an open source license. It is used within ALPAGE and (occasionally) at LORIA (Nancy) and at University of Pennsylvania.

The current version adds the notion of namespace, to get more compact and less error-prone meta-grammars. It also provides other extensions of the standard notion of Meta-Grammar in order to generate very compact TAG grammars. These extensions include the notion of *guarded nodes*, i.e. nodes whose existence and non-existence depend on the truth value of a guard, and the use of the regular operators provided by DYALOG on nodes, namely disjunction, interleaving and Kleene star. The current release provides a dump/restore mechanism for faster compilations on incremental changes of a meta-grammars.

The current version of *mgcomp* has been used to compile a wide coverage Meta-Grammar FRMG (version 2.0.1) to get a grammar of around 200 TAG trees [12]. Without the use of guarded nodes and regular operators, this grammar would have more than several thousand trees and would be almost intractable. FRMG has been packaged and is freely available.

To ease the design of meta-grammars, a set of tools have been implemented, mostly by Éric Villemonte De La Clergerie, and collected in *MGTOOLS* (version **2.2.2**). This package includes a converter from a compact format to a XML pivot format, an Emacs mode for the compact and XML formats, a graphical viewer interacting with Emacs and XSLT stylesheets to derive HTML views.

The various tools on Metagrammars are available on Inria GForge. FRMG is used directly or indirectly (through a Web service or by requiring parsed corpora) by several people and actions (ANR Rhapsodie, ANR Chronoline, ...)

5.4. The Bonsai PCFG-LA parser

Participants: Marie-Hélène Candito [correspondant], Djamé Seddah, Benoît Crabbé.

Web page:

http://alpage.inria.fr/statgram/frdep/fr_stat_dep_parsing.html

Alpage has developed as support of the research papers [60], [53], [54], [11] a statistical parser for French, named Bonsai, trained on the French Treebank. This parser provides both a phrase structure and a projective dependency structure specified in [4] as output. This parser operates sequentially: (1) it first outputs a phrase structure analysis of sentences reusing the Berkeley implementation of a PCFG-LA trained on French by Alpage (2) it applies on the resulting phrase structure trees a process of conversion to dependency parses using a combination of heuristics and classifiers trained on the French treebank. The parser currently outputs several well known formats such as Penn treebank phrase structure trees, Xerox like triples and CONLL-like format for dependencies. The parsers also comes with basic preprocessing facilities allowing to perform elementary sentence segmentation and word tokenisation, allowing in theory to process unrestricted text. However it is believed to perform better on newspaper-like text. See 6.12 for recent work and results involving Bonsai.

The parser is available under a GPL license.

5.5. The MICA parser

Participants: Benoît Sagot [correspondant], Pierre Boullier.

Web page:

<http://mica.lif.univ-mrs.fr/>

MICA (Marseille-Inria-Columbia- AT&T) is a freely available dependency parser [48] currently trained on English and Arabic data, developed in collaboration with Owen Rambow and Daniel Bauer (Columbia University) and Srinivas Bangalore (AT&T). MICA has several key characteristics that make it appealing to researchers in NLP who need an off-the-shelf parser, based on Probabilistic Tree Insertion Grammars and on the SYNTAX system. MICA is fast (450 words per second plus 6 seconds initialization on a standard high-end machine) and has close to state-of-the-art performance (87.6% unlabeled dependency accuracy on the Penn Treebank).

MICA consists of two processes: the supertagger, which associates tags representing rich syntactic information with the input word sequence, and the actual parser, based on the Inria SYNTAX system, which derives the syntactic structure from the n -best chosen supertags. Only the supertagger uses lexical information, the parser only sees the supertag hypotheses.

MICA returns n -best parses for arbitrary n ; parse trees are associated with probabilities. A packed forest can also be returned.

5.6. Alpage's linguistic workbench, including SxPipe

Participants: Benoît Sagot [correspondant], Rosa Stern, Marion Baranes, Damien Nouvel, Virginie Mouilleron, Pierre Boullier, Éric Villemonte de La Clergerie.

See also the web page <http://lingwb.gforge.inria.fr/>.

Alpage's linguistic workbench is a set of packages for corpus processing and parsing. Among these packages, the SxPipe package is of a particular importance.

SxPipe [80] is a modular and customizable chain aimed to apply to raw corpora a cascade of surface processing steps. It is used

- as a preliminary step before Alpage's parsers (e.g., FRMG);
- for surface processing (named entities recognition, text normalization, unknown word extraction and processing...).

Developed for French and for other languages, SxPipe includes, among others, various named entities recognition modules in raw text, a sentence segmenter and tokenizer, a spelling corrector and compound words recognizer, and an original context-free patterns recognizer, used by several specialized grammars (numbers, impersonal constructions, quotations...). In 2012, SxPipe has received a renewed attention in four directions:

- Support of new languages, and most notably German (although this is still at a very preliminary stage of development);
- Analysis of unknown words, in particular in the context of the ANR project EDyLex and of the collaboration with *viavoo*; this involves in particular (i) new tools for the automatic pre-classification of unknown words (acronyms, loan words...) (ii) new morphological analysis tools, most notably automatic tools for constructional morphology (both derivational and compositional), following the results of dedicated corpus-based studies (see 6.2 for new results);
- Development of new local grammars for detecting new types of entities and improvement of existing ones, in the context of the PACTE project (see 6.7 for new results).

5.7. MElt

Participants: Benoît Sagot [correspondant], Pierre Magistry.

MElt is a part-of-speech tagger, initially developed in collaboration with Pascal Denis (Magnet, Inria — then at Alpage), which was trained for French (on the French TreeBank and coupled with the *Lefff*), also trained on English [63], Spanish [69], Italian [94], German, Dutch, Polish, Kurmanji Kurdish [104] and Persian [89], [90]. It is state-of-the-art for French.

It is now able to handle noisy corpora (French and English only).

MElt also includes a lemmatization post-processing step.

A specific effort has been made towards the usability of MElt by linguists. In particular, a training session has been organized, and a user guide has been written.

Moreover, a preliminary version of MElt which accepts input DAGs has been developed.

MElt is distributed freely as a part of the Alpage linguistic workbench.

5.8. The Alexina framework: the Lefff syntactic lexicon, the Aleda entity database and other Alexina resources

Participants: Benoît Sagot [correspondant], Laurence Danlos.

See also the web page <http://gforge.inria.fr/projects/alexina/>.

Alexina is Alpage's Alexina framework for the acquisition and modeling of morphological and syntactic lexical information. The first and most advanced lexical resource developed in this framework is the *Lefff*, a morphological and syntactic lexicon for French.

Historically, the *Lefff* 1 was a freely available French morphological lexicon for verbs that has been automatically extracted from a very large corpus. Since version 2, the *Lefff* covers all grammatical categories (not just verbs) and includes syntactic information (such as subcategorization frames); Alpage's tools, including Alpage's parsers, rely on the *Lefff*. The version 3 of the *Lefff*, which has been released in 2008, improves the linguistic relevance and the interoperability with other lexical models.

Other Alexina lexicons exist, at various stages of development, in particular for Spanish (the *Leffe*), Polish, Slovak, English, Galician, Persian, Kurdish, Italian, German, as well as for Latin verbs and a subset of Maltese and Khaling verbs. These lexicons are used in various tools, including instances of the MElt POS-tagger, and for studies in quantitative morphology.

Alexina also hosts *Aleda* [98], [88] a large-scale entity database currently developed for French but under development for English, Spanish and German, extracted automatically from Wikipedia and Geonames. It is used among others in the SxPipe processing chain and its NP named entity recognition, as well as in the NOMOS named entity linking system.

5.9. The free French wordnet WOLF

Participants: Benoît Sagot [correspondant], Sarah Beniamine.

The WOLF (Wordnet Libre du Français) is a wordnet for French, i.e., a lexical semantic database. The development of WOLF started in 2008 [82], [83]. At this time, we focused on benefiting from available resources of three different types: general and domain-specific bilingual dictionaries, multilingual parallel corpora and Wiki resources (Wikipedia and Wiktionaries). This work was achieved in a large part in collaboration with Darja Fišer (University of Ljubljana, Slovenia), in parallel with the development of a free Slovene wordnet, sloWNet. However, it was also impacted by specific collaborations, e.g., on adverbial synsets [84].

In 2013, a beta version of a new version of WOLF (version 1.0b2) was published, which integrates and extends the various efforts performed and published somewhat independently in 2012.

The WOLF is freely available under the Cecill-C license. It has already been used in various experiments, within and outside Alpage.

5.10. OGRE (Optimized Graph Rewriting Engine)

Participants: Corentin Ribeyre [correspondant], Djamé Seddah, Éric Villemonte de La Clergerie, Marie-Hélène Candito.

OGRE (Optimized Graph Rewriting Engine) is a graph rewriting system specifically designed for manipulating linguistic trees and graphs [78]. It relies on a rule specification language for expressing graph rewriting patterns. The transformation is performed in two steps:

1. First, the system performs simple transformations following the rewriting patterns;
2. Second, constraints can be applied on edges, which applies transformations depending on their environment that are propagated while all constraints are satisfied.

The system has been designed for the analysis and manipulation of attributed oriented and multi-relational graphs.

Web site: <http://www.corentinribeyre.fr/projects/view/OGRE>

5.11. Automatic construction of distributional thesauri

Participants: Marie-Hélène Candito [correspondant], Enrique Henestroza Anguiano.

FREDIST is a freely-available (LGPL license) Python package that implements methods for the automatic construction of distributional thesauri.

We have implemented the context relation approach to distributional similarity, with various context relation types and different options for weight and measure functions to calculate distributional similarity between words. Additionally, FREDIST is highly flexible, with parameters including: context relation type(s), weight function, measure function, term frequency thresholds, part-of-speech restrictions, filtering of numerical terms, etc.

Distributional thesauri for French are also available, one each for adjectives, adverbs, common nouns, and verbs. They have been constructed with FREDIST and use the best settings obtained in an evaluation. We use the *L'Est Republicain* corpus (125 million words), *Agence France-Presse* newswire dispatches (125 million words) and a full dump of the French Wikipedia (200 million words), for a total of 450 million words of text.

5.12. Tools and resources for time processing

Participant: Laurence Danlos [correspondant].

Alpage developed the *French TimeBank*, a freely-available corpus annotated with ISO-TimeML-compliant temporal information (dates, events and relations between events) [1].

5.13. LexViz

Participants: Mikael Morardo [maintainer], Éric Villemonte de La Clergerie.

In the context of the industrial collaboration of ALPAGE with the company Lingua & Machina, we have extended their WEB platform Libellex with a new component used to visualize and collaboratively validate lexical resources. In particular, this extension is used to manage terminological lists and lexical networks. The implemented graph-based representation has proved to be intuitive and quite useful for navigating in such large lexical resources (on the order to 10K to 100K entries).

5.14. Mgwiki

Participants: Paul Bui Quang [maintainer], Éric Villemonte de La Clergerie.

In the context of Inria ADT Mgwiki, Paul Bui Quang has developed a linguistic wiki that may be used to discuss linguistic phenomena with the possibility to add annotated illustrative sentences. The work is essentially devoted to the construction of an instance for documenting and discussing FRMG, with the annotations of the sentences automatically provided by parsing them with FRMG. This instance also offers the possibility to parse small corpora with FRMG and an interface of visualization of the results.

Another instance was deployed for managing the annotation guide for the Deep version of the Sequoia treebank, confirming the potential of the notion of linguistic wiki

5.15. NewsProcess

Participants: Éric Villemonte de La Clergerie [maintainer], Damien Nouvel.

NewsProcess is an HTTP-based service that may be used to process AFP news through the Alpage Processing Chain, in order to extract information, in particular citations. The chain has been completed to track the emergence of new words in the news.

In the context on ANR EdyLex, a new version of NewsProcess has been designed for processing AFP news wires and extracting information about unknown words (see [6.2](#))

5.16. System EasyRef

Participant: Éric Villemonte de La Clergerie [maintainer].

A collaborative WEB service EASYREF has been developed, in the context of ANR action Passage, to handle syntactically annotated corpora. EASYREF may be used to view annotated corpus, in both EASY or PASSAGE formats. The annotations may be created and modified. Bug reports may be emitted. The annotations may be imported and exported. The system provides standard user right management. The interface has been designed with the objectives to be intuitive and to speed edition.

EASYREF relies on an Model View Controller design, implemented with the Perl Catalyst framework. It exploits WEB 2.0 technologies (i.e. AJAX and JavaScript).

Version 2 has been used by ELDA and LIMSI to annotate a new corpus of several thousands words for the former ANR projectPASSAGE.

EASYREF is maintained under Inria GForge.

PANAMA Project-Team

5. Software and Platforms

5.1. MPTK: the Matching Pursuit Toolkit

Participants: Rémi Gribonval [contact person], Jules Espiau de Lamaestre.

The Matching Pursuit ToolKit (MPTK) is a fast and flexible implementation of the Matching Pursuit algorithm for sparse decomposition of monophonic as well as multichannel (audio) signals. MPTK is written in C++ and runs on Windows, MacOS and Unix platforms. It is distributed under a free software license model (GNU General Public License) and comprises a library, some standalone command line utilities and scripts to plot the results under Matlab. This software has been registered at the APP (Agence de Protection des Programmes).

<http://mptk.gforge.inria.fr>, <http://mptk.irisa.fr>

5.2. FASST: a Flexible Audio Source Separation Toolbox

Participants: Nancy Bertin, Frédéric Bimbot.

Emmanuel Vincent [contact person]

FASST is a Flexible Audio Source Separation Toolbox, designed to speed up the conception and automate the implementation of new model-based audio source separation algorithms.

FASST is currently being developed jointly with the PAROLE team in Nancy and the TEXMEX team in Rennes through an Inria funded ADT (Action de Développement Technologique). The first implementation is in Matlab. <http://bass-db.gforge.inria.fr/fasst/>

5.3. NACHOS: Nearfield Acoustic HOlography with Sparse regularization

Participants: Nancy Bertin [contact person], Rémi Gribonval.

The software and associated database were developed within the ANR ECHANGE project, with the participation of Gilles Chardon, Laurent Daudet, François Ollivier and Antoine Peillot.

NACHOS (Nearfield Acoustic HOlography with Sparse regularization) is a downloadable companion software for the journal paper [79], distributed to comply with the "reproducible research" principle. It performs the reconstruction of operational deflection shapes of a vibrating structure, from acoustic measurements of the generated sound field. The software consists in Matlab source code, and automatically downloads the needed database. It allows to reproduce all results and figures of the paper, and to experiment some additional settings. It is distributed under GPL 3.0 license. Inter Deposit Digital Numbers: IDDN.FR.001.420023.000.S.P.2013.000.31235 (NACHOSDB) % IDDN.FR.001.420023.000.S.P.2013.000.31235 (NACHOS).

<http://exchange.inria.fr/nah>.

PAROLE Project-Team

5. Software and Platforms

5.1. WinSnoori

WinSnoori is a speech analysis software that we have been developing for 15 years. It is intended to facilitate the work of the scientist in automatic speech recognition, phonetics or speech signal processing. Basic functions of WinSnoori enable several types of spectrograms to be calculated and the fine edition of speech signals (cut, paste, and a number of filters) as the spectrogram allows the acoustical consequences of all the modifications to be evaluated. Beside this set of basic functions, there are various functionalities to annotate phonetically or orthographically speech files, to extract fundamental frequency, to pilot the Klatt synthesizer and to utilize PSOLA resynthesis.

The current version of WinSnoori is available on <http://www.winsnoori.fr>.

5.2. JSnoori

JSnoori is written in Java and uses signal processing algorithms developed within WinSnoori software with the double objective of being a platform independent signal visualization and manipulation tool, and also for designing exercises for learning the prosody of a foreign language. JSnoori thus focused the calculation of F0, the forced alignment of non native English uttered by French speakers and the correction of prosody parameters (F0, rhythm and energy). Since phonetic segmentations and annotations play a central role in the derivation of diagnosis concerning the realization of prosody by learners, several tools have been incorporated to segment and annotate speech. In particular, a complete phonetic keyboard is available, several kinds of annotation can be used (phonemes, syllables and words) and forced alignment can exploit variants to cope with non native accents. In addition, JSnoori offers real time F0 calculation which can be useful from a pedagogical point of view.

5.3. Xarticulators

Xarticulators software is intended to delineate contours of speech articulators in X-ray images, to construct articulatory models and to synthesize speech from X-ray films. This software provide tools to track contours automatically, semi-automatically or by hand, to make the visibility of contours easier, to add anatomical landmarks to speech articulators and to synchronize images together with the sound.

It also enables the construction of adaptable linear articulatory models from the X-ray images.

This year we particularly worked on the possibility of synthesizing speech from X-ray images. We thus substantially improved algorithms used to compute the centerline of the vocal tract in order to segment the vocal tract into elementary tubes approximating the propagation of a one-dimensional wave. We also developed time patterns used to synthesize sequences of voiceless consonants and vowels (VCV). In addition we also added the possibility of processing digitized manual delineation results made on sheet of papers in the seventies.

5.4. SUBWEB

We published in 2007 a method which allows to align sub-titles comparable corpora [94]. In 2009, we proposed an alignment web tool based on the developed algorithm. It allows to: upload a source and a target files, obtain an alignment at a sub-title level with a verbose option, and a graphical representation of the course of the algorithm. This work has been supported by CPER/TALC/SUBWEB ².

²<http://wikitalc.loria.fr/dokuwiki/doku.php?id=operations:subweb>

5.5. ANTS

The aim of the Automatic News Transcription System (ANTS) is to transcribe radio or TV shows. ANTS is composed of several stages. The first processing steps aim at splitting the audio stream into homogeneous segments of a manageable size and at identifying the segment characteristics in order to allow the use of specific algorithms or models according to the nature of the segment. This includes broad-band/narrow-band speech segmentation, speech/music classification, speaker segmentation and clustering, detection of silences/breathing segments and generally speaker gender classification.

Each segment is then decoded using a large vocabulary continuous speech recognition engine, either the Julius engine or the Sphinx engine. The Julius engine operates in two passes: in the first pass, a frame-synchronous beam search algorithm is applied on a tree-structured lexicon assigned with bigram language model probabilities. The output of this pass is a word-lattice. In the second pass, a stack decoding algorithm using a trigram language model gives the N-best recognition sentences. The Sphinx engine processes the speech input segment in a single forward pass using a trigram language model.

Further processing passes are usually run in order to apply unsupervised adaptation processes on the feature computations (VTLN: vocal tract length normalization) and/or on the model parameters (MLLR: maximum likelihood linear regression), or to use speaker adaptive training (SAT) based models. Moreover decoding results of both systems can be efficiently combined for improved decoding performance.

The latest version which relies on a perl script exploits the multiple CPUs available on a computer to reduce the processing time, and runs on both a stand alone linux machine and on the cluster.

5.6. CoALT

CoALT (Comparing Automatic Labeling Tools) compares two automatic labelers or two speech-text alignment tools, ranks them and displays statistics about their differences. The main feature of our software is that a user can define its own criteria for evaluating and comparing two speech- text alignment tools. With CoALT, a user can give more importance to either phoneme labels or phoneme boundaries because the CoALT elastic comparison algorithm takes into account time boundaries. Moreover, by providing a set of phonetic rules, a user can define the allowed discrepancies between the automatic labeling result and the hand-labeling one.

5.7. TTS SoJA

TTS SoJA (Speech synthesis platform in Java) is a software for text-to-speech synthesis. The aim of this software is to provide a toolkit to test some steps of natural language processing and to provide a whole system of TTS based on non uniform unit selection algorithm. The software performs all steps from the text to the speech signal. Moreover, it provides a set of tools to elaborate a corpus for a TTS system (transcription alignment, ...). Currently, the corpus contains 1800 sentences (about 3 hours of speech) recorded by a female speaker.

Most of the modules are developed in Java. Some modules are in C. The platform is designed to make easy the addition of new modules. The software runs under Windows and Linux (tested on Mandriva, Ubuntu). It can be launch with a graphical user interface or directly integrated in a Java code or by following the client-server paradigm.

The software license should easily allow associations of impaired people to use the software. A demo web site has been built: <http://soja-tts.loria.fr>

5.8. JCorpusRecorder

JCorpusRecorder is a software for the recording of audio corpora. It provides a easy tool to record with a microphone. The audio input gain is controlled during the recording. From a list of sentences, the output is a set of wav files automatically renamed with textual information given in input (nationality, speaker language, gender...). An easy syntactic tagging allows displaying a textual/visual/audio context of the sentence to pronounce. This software is suitable for recording sentences with information to guide the speaker. The sentences can be presented randomly.

The software is now developed in Java (since 2013). It is currently used for the recording of sentences in several projects (including IFCASL).

5.9. VisArtico

VisArtico is intended to visualize articulatory data acquired using an articulograph [97]. It is intended for researchers that need to visualize data acquired from the articulograph with no excessive processing. It is well adapted to the data acquired using the AG500 and AG501 (developed by Carstens Medizinelektronik GmbH), and the articulograph NDI Wave, developed by Northern Digital Inc.

The software allows displaying the positions of the sensors that are simultaneously animated with the speech signal. It is possible to display the tongue contour and the lips contour. The software helps to find the midsagittal plane of the speaker and find the palate contour. In addition, VisArtico allows labeling phonetically the articulatory data.

All this information is very useful to researchers working in the field of speech production, as phoneticians for instance. VisArtico provides several possible views: (1) temporal view, (2) 3D spatial view and (3) 2D midsagittal view. In the temporal view, it is possible to display different articulatory trajectories in addition to the acoustic signal and eventually labels. The midsagittal view can display the tongue contour, the jaw, the lips and the palate.

VisArtico provides several tools to help to improve the quality of interpreting the data. It is a cross-platform software as it is developed in JAVA and does not need any additional external library or framework. It was tested and worked on Windows, Mac OS, and Linux. It should work on any system having JAVA installed. VisArtico is freely distributed via a dedicated website <http://visartico.loria.fr>.

5.10. FASST

The Flexible Audio Source Separation Toolbox (FASST) is a toolbox for audio source separation (<http://bass-db.gforge.inria.fr/fasst/>). It aims to become the reference software for research and applications of audio source separation. Its unique feature is the possibility for users to specify easily a suitable algorithm for their use case thanks to the general modeling and estimation framework. Besides, it forms the basis of most of our current research in audio source separation, some of which may be incorporated into future versions of the software.

SÉMAGRAMME Project-Team

5. Software and Platforms

5.1. Leopard

Participants: Bruno Guillaume [correspondent], Guy Perrier, Tatiana Ekeinhor.

5.1.1. Software description

Leopard is a parser for natural languages which is based on the formalism of Interaction Grammars [40]. It uses a parsing principle, called “electrostatic parsing” which consists in neutralizing opposite polarities. A positive polarity corresponds to an available linguistic feature and a negative one to an expected feature.

Parsing a sentence with an Interaction Grammar consists in first selecting a lexical entry for each of its words. A lexical entry is an underspecified syntactic tree, a tree description in other words. Then, all selected tree descriptions are combined by partial superposition guided by the aim of neutralizing polarities: two opposite polarities are neutralized by merging their support nodes. Parsing succeeds if the process ends with a minimal and neutral tree. As IGs are based on polarities and under-specified trees, Leopard uses some specific and non-trivial data-structures and algorithms.

The electrostatic principle has been intensively considered in Leopard. The theoretical problem of parsing IGs is NP-complete; the nondeterminism usually associated to NP-completeness is present at two levels: when a description for each word is selected from the lexicon, and when a choice of which nodes to merge is made. Polarities have shown their efficiency in pruning the search tree:

- In the first step (tagging the words of the sentence with tree descriptions), we forget the structure of descriptions, and only keep the bag of their features. In this case, parsing inside the formalism is greatly simplified because composition rules reduce to the neutralization of a negative feature-value pair $f \leftarrow v$ by a dual positive feature-value pair $f \rightarrow v$. As a consequence, parsing reduces to a counting of positive and negative polarities present in the selected tagging for every pair (f, v) : every positive occurrence counts for +1 and every negative occurrence for -1, the sum must be 0.
- Again in the tagging step, original methods were developed to filter out bad taggings. Each unsaturated polarity p in the grammar induces constraints on the set of contexts in which it can be used: the unsaturated polarity p must find a *companion* (i.e. a tree description able to saturated it); and the set of companions for the polarity p can be computed statically from the grammar. Each lexical selection which contains an unsaturated polarity without one of its companions can be safely removed.
- In the next step (node-merging phase), polarities are used to cut off parsing branches when their trees contain too many non neutral polarities.

5.1.2. Current state of the implementation

Leopard is presented and documented at <http://leopard.loria.fr>; an online demonstration page can be found at <http://leopard.loria.fr/demo>.

It is open-source (under the CECILL License <http://www.cecill.info>) and it is developed using the InriaGforge platform (<http://gforge.inria.fr/projects/semagramme/>)

The main features of current software are:

- automatic parsing of a sentence or a set of sentences,
- dependency and parse-tree representation of sentences,
- interactive parsing (the user chooses the couple of nodes to merge),
- visualization of grammars produced by XMG-2 or of sets of description trees associated to some word in the linguistic resources.

One of the difficulties with symbolic parsing is that several solutions can be produced for a single sentence and we want to be able to rank them. Tatiana Ekeinhor, during her second year Master Internship (from February to June 2013), implemented a ranker based on statistical techniques. Using the Sequoia TreeBank as a training corpus, she obtained an improvement of the system compared to the handcrafted rules.

5.2. ACG Development Toolkit

Participants: Sylvain Pogodalla [correspondent], Philippe de Groot.

In order to support the theoretical work on ACG, we have been developing a support system. The objectives of such a system are twofold:

1. To make possible to implement and experiment grammars the modeling of linguistic phenomena.
2. To make possible to implement and experiment results related to the ACG formalisms. Such results can concern parsing algorithms, type extensions, language extensions, etc.

The ACG Development toolkit development effort is part of the POLYMNIE project (see Section 7.2.1.1). It will support the experimentation and evaluation parts of the project.

The current version of the ACG development toolkit prototype¹ issues from a first release published in October 2008. Further releases have been published before the ESSLLI 2009 course on ACG. It focuses on providing facilities to develop grammars. To this end, the type system currently implemented is the linear core system plus the (non-linear) intuitionistic implication, and a special attention has been paid to type error management. As a major limitation, this version only considers transformation from abstract terms to object terms, and not the other way around.

The prototype now enables the transformation from the object terms to the abstract terms. The parsing algorithm follows [43]'s method which is being implemented for second-order ACGs. It is based on a translation of ACG grammars into Datalog programs and is well-suited to fine-grained optimization.

However, since we're interested not only by recognizability (hence whether some fact is provable) but also by the parsing structure (hence the proof), the Datalog solver has been adapted to produce not only yes/no answer to queries, but also all the proofs of the answers to the queries. The next steps concern optimization and efficiency. Note however that in the general case, the decidability of translating an object term to an abstract one is still an open problem.

5.3. Grew

Participants: Bruno Guillaume [correspondent], Guy Perrier.

Graph rewriting, Interface syntaxe-sémantique

Grew is a Graph Rewriting tools dedicated to applications in NLP. It is freely-available (from the page <http://grew.loria.fr>) and it is developed using the InriaGforge platform (<http://gforge.inria.fr/projects/semagramme/>)

We list below some of the major specificities of the GREW software.

- Graph structures can use a build-in notion of feature structures.
- The left-hand side of a rule is described by a graph called a pattern; injective graph morphisms are used in the pattern matching algorithm.
- Negative pattern can be used for a finer control on the left-hand side of rules.
- The right-hand side of rules is described by a sequence of atomic commands that describe how the graph should be modified during the rule application.
- Rules can be parametrized by lexical information.
- Filters can be used at the output of each module to control the structure produced are well-formed.

¹ Available at <http://acg.gforge.inria.fr> with a CeCILL license.

- Subset of rules are grouped in modules; the full rewriting process being a sequence of module applications.
- The Grew software has support both for confluent and non-confluent modules; when a non-confluent modules is used, all normal forms are returned and then ambiguity is handled in a natural way.
- Grew can be used on Corpus mode with statistics about rules usage or with an a Graphical User Interface which can show all intermediate graphs used during the rewriting process (useful either to debug rewriting system or for demonstrations).

The Grew software was used for several kind of applications manipulating syntactic and/or semantic graph representations. It was used to build DMRS semantic representation from syntactic dependency trees in the French TreeBank [51].

More recently, it was used in the project “Deep Syntax Annotation of the Sequoia French Treebank”. First, it was used as a pre-annotation tool and; second, it is used to detect ill-formed structures that don’t fit the annotation guide requirement.

5.4. Other developments

Participants: Bruno Guillaume [correspondent], Maxime Amblard [correspondent].

Concordancer, Dependencies, Graphical tools Other peripheral developments of the team are available either as web service or as downloadable code:

- A concordancer named CONDOR which is usable online: <http://condor.loria.fr>. With Condor, it is possible to search for all inflexions (given by a lexicon) of some lemma; it is possible to search for a couple of lemmas to find collocations.
- A program (named DEP2PICT) to build graphical representations (PNG, SVG or PDF) of dependency structures. It is presented in <http://dep2pict.loria.fr>; it is usable online <http://dep2pict.loria.fr/demo>.
- a management chain of the transcriptions of interviews for the SLAMproject. including the production of a full anonymized randomized version of the resources.
- A program which use Distagger and propose different analyze of the repartition of disfluencies.

COPRIN Project-Team

5. Software and Platforms

5.1. Introduction

Software development is an essential part of the research done by COPRIN since a large part of our methods can only be validated experimentally (both for our numerical experiments and in robotics). Software developments follow various directions:

1. interval arithmetic: although we do not plan to work in this very specialized area (we generally rely on existing packages) interval arithmetic is an important part of our interval analysis algorithms and we may have to modify the existing packages so as to deal, in particular, with multi-precision and arithmetic extensions
2. interval analysis libraries: we daily use the ALIAS library that has been designed in the project and is still under development. A long term work is to develop a generic programming framework that allows for modularity and flexibility, with the objectives of testing new functionalities easily and building specific solvers by a simple juxtaposition of existing modules
3. interface to interval analysis: in our opinion interval analysis software must be available within general purpose scientific software (such as Maple, Mathematica, Scilab) and not only as a stand-alone tool. Indeed most end-users are reluctant to learn a new programming language just to solve problems that are only small elements of a more general problem. Furthermore interval analysis efficiency may benefit from the functionalities available in the general purpose scientific software.

5.2. Interval analysis libraries

5.2.1. ALIAS

Participants: David Daney, Jean-Pierre Merlet [correspondant], Odile Pourtallier.

The ALIAS library (*Algorithms Library of Interval Analysis for Systems*), whose development started in 1998, is a collection of procedures based on interval analysis for systems solving and optimization.

ALIAS is made of two parts:

- ALIAS-C++: the C++ library (87 000 code lines) which is the core of the algorithms
- ALIAS-Maple: the Maple interface for ALIAS-C++ (55 000 code lines). This interface allows one to specify a solving problem within Maple and get the results within the same Maple session. The role of this interface is not only to generate the C++ code automatically, but also to perform an analysis of the problem in order to improve the efficiency of the solver. Furthermore, a distributed implementation of the algorithms is available directly within the interface.

Although these libraries are intended to be used within the project-team they can be freely downloaded as a library file (but the user may introduce its own code in several part of the package) and has been used for example at LIRMM and IRCCyN.

IMARA Project-Team

5. Software and Platforms

5.1. MELOSYM

Participants: Fawzi Nashashibi [correspondant], Benjamin Lefaudeux, Paulo Lopes Resende.

MELOSYM is the acronym for “**M**odélisation de l’**E**nvironnement et **L**ocalisation en temps réel pour un **S**ystème **M**obile autonome ou pas, fondé sur des données du capteur laser”. This is a SLAM based algorithm for the environment mapping and vehicle localization in real-time using laser data. The particularity of the algorithm is its hierarchical approach that improves the accuracy of the system and speeds up the computations. Version 3 is under edition. It runs now in standalone mode without the use of RTMaps software libraries.

- Version: V2

5.2. Stereoloc-3D

Participants: Benjamin Lefaudeux, Fawzi Nashashibi [correspondant].

This software is a stereovision based system capable of performing a vehicle or robot ego-localization and 3D environment mapping in real-time. It has also the capability to ensure mobile objects detection and tracking. A new updated version has been released and tested on a mobile platform.

- Version: V1

5.3. Fuzzy logic tool

Participant: Joshué Pérez Rastelli [correspondant].

A fuzzy logic module has been implemented to translate human knowledge to driverless control processes, considering risk/warning situation. Fuzzy logic techniques have been widely implemented in different industrial process in the last decade. For this reason, many libraries, mainly developed in C++, are easily found in the literature. The goal is to achieve the autonomous driving of the vehicle using simple sentences defined in a rule base. Then, it is just necessary to define the input and output membership functions. Two modules based on fuzzy logic libraries were created. One of them was developed in order to compare the classic controller of a previous work with a fuzzy controller to improve the lateral control tracking previously developed. Moreover, another module to warn speed references at intersections with traffic lights was done in the framework of the project CoDrive. The idea is that the vehicle is able to know at which speed it must travel to avoid abrupt braking and save fuel.

5.4. Dynamic path generation

Participant: Joshué Pérez Rastelli [correspondant].

An algorithm for dynamic path generation in urban environments is presented, taking into account structural and sudden changes in straight and bend segments (e.g. roundabouts and intersections). The results present some improvements in path generation (previously hand plotted) considering parametric equations and continuous-curvature algorithms, which guarantees a comfortable lateral acceleration. This work is focused in a smooth and safe path generation using road and obstacle detection information. Finally, some simulation results show a good performance of the algorithm using different ranges of urban curves.

5.5. V2ProVu

Participants: Pierre Merdrignac, Oyunchimeg Shagdar [correspondant].

A Java-based software is developed to enable direct Wi-Fi communications between devices, especially between vehicle on-board communication devices and pedestrian hand-held devices (e.g., tablets). The software includes an algorithm that calculates vehicle-to-pedestrian collision risk and GUI, for hazard alarming.

5.6. Network Selector

Participant: Oyunchimeg Shagdar [correspondant].

An OSGi based software is developed under the scope of SCORE@F project. The software has the functionality of switching between Geo- and IP-networks in vehicular communications allowing e.g., Cooperative Awareness Messages (CAM) as well as Decentralized Event Notification Messages (DENM) being able distributed over one or both of the Geo- and IP-networks.

5.7. FAC-CM

Participant: Manabu Tsukada [correspondant].

An OSGi based software is developed under the scope of SCORE@F project. The software allows information exchange between Facilities and Management entities of ITS stations (e.g., vehicle on-board communication device).

E-MOTION Project-Team

4. Software and Platforms

4.1. PROTEUS Software

Participants: Amaury Nègre, Juan Lahera-Perez.

This toolkit offers a automatic mobile robot driver, some sensors drivers sensors as Sick laser, GPS, motion tracker, mono or stereo camera), and a 3D Simulator.

The latest developments have been focuses on the robotics simulator. This simulator is based on the simulation and 3D rendering engine “mgEngine“ (<http://mgengine.sourceforge.net/>) embedded with the physics engine “bullets physics” (<http://bulletphysics.org>) for realistic robot dynamic simulation.

We also worked on the interface with the robotics middleware “ROS“ (<http://www.ros.org>) in order to offer interoperability with many robotics applications.

The simulator is now fully integrated with the robotics middleware "ROS" (<http://www.ros.org>) which allow interoperability with a large set of robotics applications and visualization tools. This software is developed in C++ and the simulator operates with the Lua scripting language. The simulation software is used in the ANR Proteus (<http://www.anr-proteus.fr>), as a simulation engine for the PROTEUS Toolkit.

- Version: 2.0
- APP:IDDN.FR.001.510040.000.S.P.2005.000.10000
- Programming language: C/C++, Lua

4.2. AROSDYN

Participants: Igor Paromtchik, Mathias Perrollaz, Alexandros Makris, Amaury Nègre, Christian Laugier.

ArosDyn (<http://arosdyn.gforge.inria.fr/>) is a system which integrates our recently developped techniques to provide a real-time collision risk estimation in a dynamic environment. The main features of this software are:

1. The design provides high maintainability, scalability and reuseness of the models and algorithms.
2. The software has a user interface (UI) which is user-friendly.
3. The software facilitates the parameter tuning of the models.
4. It uses the GPU to accelerate the computation.
5. Working together with the Hugn middleware (<http://gforge.inria.fr/projects/cycabtk>), it can run on our experimental vehicle in real-time.

Another important property of this software is a large part of the computation task executed on GPU. As the processing of stereo image and the computaion in the BOF can be highly parallelized, we run these tasks on the GPU to improve the time performance. The GPU calculation is based on CUDA library and is carried out in an independent thread.

Furthermore, thanks to the design of the software, we can easily add new models to it and let them work together. The fast detection and tracking algorithm (FCTA) and the Gaussian process based collision assessment algorithm are added into this framework. The software is implemented on the Lexus car. In 2012, a demand for deposing the GPU BOF software to the APP is in progress.

4.3. Embedded Perception

Participants: Mathias Perrollaz, Amaury Nègre, Christian Laugier.

The method for computing occupancy grids from a stereoscopic sensor, developed in the e-motion team, has been implemented on GPU, using NVIDIA CUDA. This allows a real time implementation and an online processing within the Lexus experimental platform.

The program has been deposited to the APP in 2012, under the reference: IDDN.FR.001.270004.000.S.P.2012.000.10800

4.4. Bayesian Occupancy Filter

People involved: Kamel Mekhnacha, Tay Meng Keat Christopher, C. Laugier, M. Yguel, Pierre Bessière. The BOF toolbox is a C++ library that implements the Bayesian Occupancy Filter. It is often used for modelling dynamic environments. It contains the relevant functions for performing bayesian filtering in grid spaces. The output from the BOF toolbox are the estimated probability distributions of each cell's occupancy and velocity. Some basic sensor models such as the laser scanner sensor model or Gaussian sensor model for gridded spaces are also included in the BOF toolbox. The sensor models and BOF mechanism in the BOF toolbox provides the necessary tools for modelling dynamic environments in most robotic applications. This toolbox is patented under two patents : "Procédé d'assistance à la conduite d'un véhicule et dispositif associé" n. 0552735 (9 september 2005) and "Procédé d'assistance à la conduite d'un véhicule et dispositif associé amélioré" n. 0552736 (9 september 2005) and commercialized by ProBayes.

- Version: 1
- Patent: 0552736 (2005), 0552735 (2005)
- Programming language: C/C++

4.5. PROBT

People involved: Juan-Manuel Ahuactzin, Kamel Mekhnacha, Pierre Bessière, Emmanuel Mazer, Manuel Yguel, Christian Laugier.

ProBT is both available as a commercial product (ProBAYES.com) and as a free library for public research and academic purposes (<http://emotion.inrialpes.fr/BP/spip.php?rubrique6>). Formerly known as *OPL*, *ProBT* is a C++ library for developing efficient Bayesian software. It is available for Linux, Unix, PC Windows (Visual C++), MacOS9, MacOSX and Irix systems. The ProBT library (<http://www.probayes.com/>) has two main components: (i) a friendly Application Program Interface (API) for building Bayesian models, and (ii) a high-performance Bayesian Inference Engine (BIE) allowing to execute all the probability calculus in exact or approximate way. *ProBT* is now commercialized by our start-up *Probayes*; it represents the main Bayesian programming tool of the *e-Motion* project-team, and it is currently used in a variety of external projects both in the academic and industrial field (e.g., for the European project BACS and for some industrial applications such as Toyota or Denso future driving assistance systems).

FLOWERS Project-Team

5. Software and Platforms

5.1. Perception Tools

Participants: David Filliat [correspondant], Natalia Lyubova, Louis-Charles Caron, Alexander Gepperth.

5.1.1. Perception Abstraction Engine

Participants: David Filliat [correspondant], Natalia Lyubova.

PAE (Perception Abstraction Engine) is a C++ library developed to provide a uniform interface to existing visual feature detector such as SIFT, SURF, MSER, superpixels, etc... Its main goal is to be able to use these various feature detectors in a "bag of feature" approach for applications such as robot localisation and object recognition. Several approach are also implemented for the visual vocabularies, in particular the fast incremental vocabularies developed in the team.

The library provide common C++ interfaces to feature detectors, visual features and visual vocabularies. A factory approach make it possible to change the feature detectors and visual vocabularies types and parameters through configuration strings, without the need to recompile. Some applications are also included in the library, in particular topological robot localization (room recognition) and visual object recognition. An Urbi interface is also provided for these modules.

5.1.2. Incremental object discovery

Participants: Natalia Lyubova [correspondant], David Filliat.

This software makes it possible to detect, model and recognize objects in a scenario of interaction between a humanoid robot and a human teacher. It is based either on standard images, or on the kinect camera to take advantage of the depth information. The software is written in C++ and relies mainly on PAE and OpenCV.

The software implements several modules: candidate object segmentation based on motion information, keypoint-based object tracking, incremental object model construction integrating multiple features (keypoints + superpixels) and object categorisation based on mutual information with robot motors (making it possible to segment robot parts, objects and humans). Based on all these modules, it is possible for the robot to learn objects shown by a human partner and to improve the objects models by manipulating them when they are put in front of the robot.

5.1.3. Object recognition from a 3-D point cloud

Participants: Louis-Charles Caron [correspondant], Alexander Gepperth, David Filliat.

This software scans the 3-D point cloud of a scene to find objects and match them against a database of known objects. The process consists in 3 stages. The segmentation step finds the objects in the point cloud, the feature extraction computes discriminating properties to be used in the classification stage for object recognition.

The segmentation is based on simple assumptions about the geometry of an indoor scene. Successive RANSACs are used to find large planes, which correspond to the floor, ceiling and walls. The cloud is stripped from the points belonging to these planes. The remaining points are clustered, meaning that close-by points are considered to form a single object.

Objects are characterized by their shape and color. Color histograms and SIFT features are computed, using the PAE library, to capture the visual appearance of the objects. Their shape is encoded by computing thousands of randomly chosen SURFLET features to construct a relative frequency histogram.

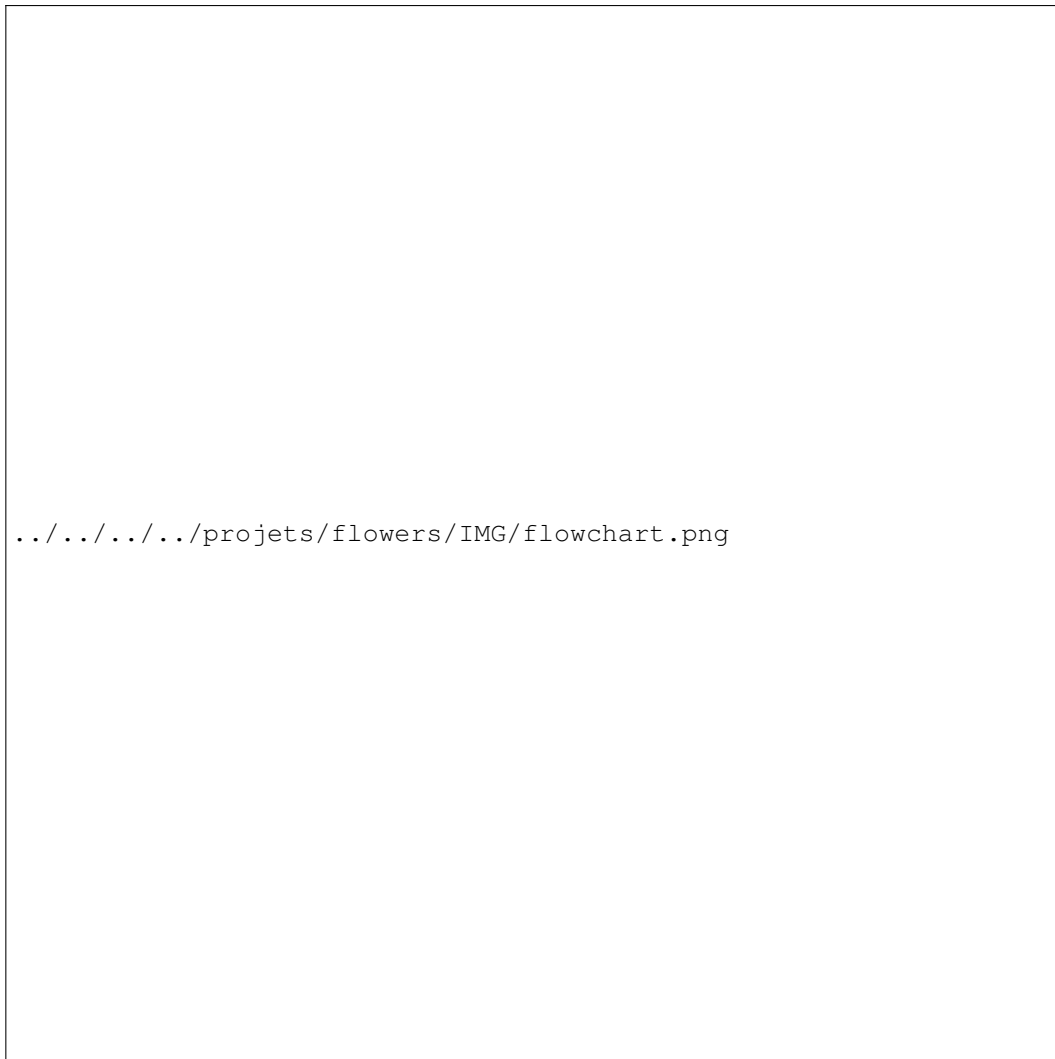


Figure 1. System Overview of the Incremental object discovery Software.

An early classification is done using each of the 3 features separately. For the color features a bag of words approach (from PAE) is used. For the shape feature, the minimum squared distance between the object's histogram and that of all objects in the database is calculated. Classification scores are then fused by a feed-forward neural network to get the final result [81].

5.1.4. PEDDETECT: GPU-accelerated person detection demo

Participant: Alexander Gepperth [correspondant].

PEDDETECT implements real-time person detection in indoor or outdoor environments. It can grab image data directly from one or several USB cameras, as well as from pre-recorded video streams. It detects multiple persons in 800x600 color images at frame rates of >15Hz, depending on available GPU power. In addition, it also classifies the pose of detected persons in one of the four categories "seen from the front", "seen from the back", "facing left" and "facing right". The software makes use of advanced feature computation and nonlinear SVM techniques which are accelerated using the CUDA interface to GPU programming to achieve high frame rates. It was developed in the context of an ongoing collaboration with Honda Research Institute USA, Inc.

5.1.5. A Python OptiTrack client

Participant: Pierre Rouanet [correspondant].

This python library allows you to connect to an OptiTrack from NaturalPoint (<http://www.naturalpoint.com/optitrack/>). This camera permits the tracking of 3D markers efficiently and robustly. With this library, you can connect to the Motive software used by the OptiTrack and retrieve the 3D position and orientation of all your tracked markers directly from python.

5.2. Datasets

5.2.1. Choreography dataset 1 and 2

Participants: Olivier Mangin [correspondant], Haylee Fogg.

These databases contain choreography motions recorded through a kinect device. In the first dataset, these motions have a combinatorial structure: from a given set of primitive dance motions, choreographies are constructed as simultaneous execution of some of these primitive motions. Primitive dance motions are chosen from a total set of 48 motions and are spanned over one or two limbs, either the legs (e.g. walk, squat), left or right arm (e.g. wave hand, punch) or both arms (e.g. clap in hands, paddle). Complex choreographies are produced as the simultaneous demonstration of two or three of these primitive motion: either one for legs and one for both arm, or one for legs and one for each arm. The dataset has been used in the experiments from [104] for studying learning techniques allowing to identify dictionaries of motion primitives, and is publicly available at https://flowers.inria.fr/choreography_database.html.

The second dataset only contains choreographies composed of a single motion. It contains 110 records of each gesture from a set of 10 simple gestures and was used in the experiments from [53]. The dataset is publicly available at <https://flowers.inria.fr/choreo2>.

5.2.2. Development on NoFish Platform

Participants: Paul Fudal [correspondant], Sao Mai Nguyen.

NoFish (fig. 2) platform is a setup used by Mai Nguyen to perform several experiments following her PhD work on social learning and intrinsic motivation. The setup consists to an ErgoRobot (fig. 3) with a fishing rod attached to the tip and a cap made with a red juggling ball. The robot is plugged on an ethernet power-switch used to turn it off if something wrong happens with the robot (for example : the robot try to go in a position he cannot reach and its motors are forcing too much). Tracking the cap is made with Full HD camera on the ceiling. At last, a video-projector prints informations on the floor (fig. 4) which helps for interactions between humans and the robots.



Figure 2. Illustration of the NoFishPlatform



Figure 3. Illustration of NoFish ErgoRobot



Figure 4. NoFish cap and informations printed on the floor

Controlling the robot is made using URBI¹ scripts allowing different control level from single motor control to pre-programmed primitive (for example : resetting the robot or make it going to its starting position). URBI is also used to perform action through the power-switch if the robot must be turned off.

Tracking the cap is made by a program written using OpenCV and keep up to date the cap's coordinates by sending them to URBI through a network socket.

An other program written in JAVA and Processing² allows to print informations on the floor; it consists of a server receiving through a network socket texts and shapes informations to print on the floor and a JAVA object which can be used with compatible software. This component were really useful for giving simple and direct information during an experiment and also during interaction between a human and the robot for socially guided experiments.

The main program is written in Matlab and includes all the explained previous components and softwares. It also includes a forward kinematic calculation module used to ensure a movement sent can be safely be played by the robot to avoid it breaking himself; this module gives step by step informations of what will happen when the robot will play the movement like, for example, if the fishing cap will touch the floor at the end or not which permits to keep the robot safe and speed up experiments by ignoring dangerous moves for the robot and useless ones for the deployed algorithm.

This setup were used by Sao Mai Nguyen to run experiments during her thesis on social learning and intrinsic motivation

5.3. Learning algorithms

5.3.1. KidLearn

Participants: Manuel Lopes [correspondant], Benjamin Clement, Pierre-Yves Oudeyer, Didier Roy.

The KidLearn software provides an Intelligent Tutoring System that optimizes teaching sequences based on the estimated level of each particular student [65]. We implemented a Game of Money that allows students, ages 7-8, to learn how to use money. It includes 3 main components: i) a webserver that handles the requests and stores the experiments in a database; ii) a GUI that provides the interface for the game; and iii) the optimization software.

Graphical interfaces in ITS can have unwanted side effects. For this reason, the interface was entirely designed with the help of a didactician, with several specific design choices motivated by pedagogical, motivational and attention requirements. For example, the interface, shown in Figure 5, is such that:

- display is as clear and simple as possible;
- there is no chronometer, so that students are not put under time pressure;
- coins and banknotes have realistic visual appearance, and their relative sizes are respected;
- display of prices use visual encodings commonly used in shops;
- the zone for receiving money is automatically cleared in case of error after the student submits it;
- automatic snapping of money and tokens icons in the reception zone, and automatic visual arrangement;
- text quantity is kept to minimum;

Four principal regions are defined in the graphical interface, as shown in Figure 5. The first is the wallet location where users can pick and drag the money to drop them on the repository location to make for the correct price. The object and the price are present in the object location, where the price can be written and/or spoken depending on the parameterization of the activity. The information location is used to display information for the learners such as extra clues when they make a mistake (for which they have to press the light bulb) and feedback. In order to improve the pedagogical success of the activity, the correct solution is presented automatically to the students if they fail to compose the correct price after 3 trials.

¹<http://www.gostai.com/products/jazz/urbi/>

²<http://processing.org>

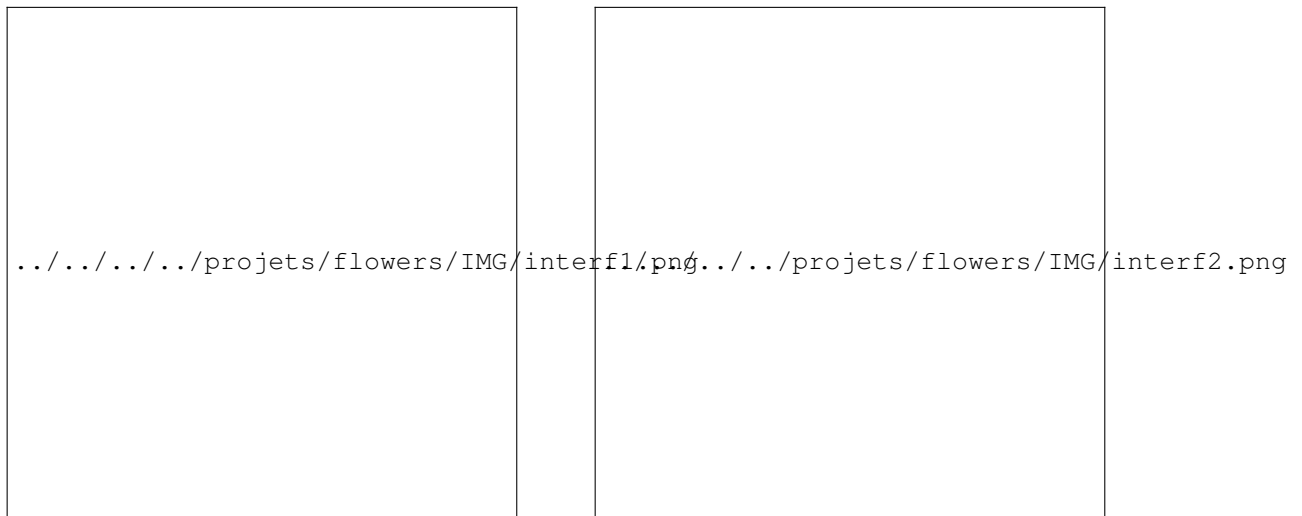


Figure 5. Example interface with four principal regions of interaction: wallet location, repository, object and price, and feedback location.

5.3.2. RLPark - Reinforcement Learning Algorithms in JAVA

Participant: Thomas Degris [correspondant].

RLPark is a reinforcement learning framework in Java. RLPark includes learning algorithms, state representations, reinforcement learning architectures, standard benchmark problems, communication interfaces for three robots, a framework for running experiments on clusters, and real-time visualization using Zephyr. More precisely, RLPark includes:

- Online Learning Algorithms: Sarsa, Expected Sarsa, Q-Learning, On-policy and off-policy Actor-Critic with normal distribution (continuous actions) and Boltzmann distribution (discrete action), average reward actor-critic, TD, TD(λ), GTD(λ), GQ(λ), TDC
- State Representations: tile coding (with no hashing, hashing and hashing with mumur2), Linear Threshold Unit, observation history, feature normalization, radial basis functions
- Interface with Robots: the Critterbot, iRobot Create, Nao, Puppy, Dynamixel motors
- Benchmark Problems: mountain car, swing-up pendulum, random walk, continuous grid world

An example of RLPark running an online learning experiment on a reinforcement learning benchmark problem is shown in Figure 6 .

RLPark was started in spring 2009 in the RLAI group at the university of Alberta (Canada) when Thomas Degris was a postdoc in this group. RLPark is still actively used by RLAI. Collaborators and users include Adam White, Joseph Modayil and Patrick Pilarski (testing) from the University of Alberta.

RLPark has been used by Richard Sutton, a professor and iCORE chair in the department of computing science at the University of Alberta, for a demo in his invited talk *Learning About Sensorimotor Data* at the Neural Information Processing Systems (NIPS) 2011 ³. Patrick Pilarski used RLPark for live demos on television (Breakfast Television Edmonton, CityTV, June 5th, 2012) and at TEDx Edmonton on Intelligent Artificial Limbs⁴. So far, RLPark has been used in more than a dozens of publications (see <http://rlpark.github.com/publications.html> for a list).

³<http://webdocs.cs.ualberta.ca/~sutton/Talks/Talks.html#sensorimotor>

⁴<http://www.youtube.com/watch?v=YPc-Ae7zqSo>

RLPark has been ported to C++ by Saminda Abeyruwan, a student of the University of Miami (United States of America). The Horde architecture in RLPark has been optimized for GPU by Clément Gehring, a student of the McGill University in Montreal (Canada).

Future developments include the implementation of additional algorithms (the Dyna architecture, back propagation in neural networks, ...). A paper is under review for the JMLR Machine Learning Open Source Software. Documentation and tutorials are included on the RLPark web site ⁵. RLPark is licensed under the open source Eclipse Public License.



Figure 6. An example of an experiment in RLPark. Zephyr displays two views of a learned weight vector, an animation of the problem, the current policy distribution learned by the algorithm and the reward obtained by the algorithm. Videos are available at: <http://rlpark.github.com>.

5.3.3. DMP-BBO Matlab library

Participant: Freek Stulp [correspondant].

The dmp_bbo (Black-Box Optimization for Dynamic Movement Primitives) Matlab library is a direct consequence of the insight that black-box optimization outperforms reinforcement learning when using policies represented as Dynamic Movement Primitives. It implements several variants of the PI^{BB} algorithm for direct policy search. It is currently being used and extended by several FLOWERS members (Manuel Lopes, Clément Moulin-Frier) and external collaborators (Jonas Buchli, Hwangbo Jemin of ETH Zurich). This code was used for the following publications: [130], [127], [128].

⁵<http://rlpark.github.com>

5.3.4. Self-calibration BCI - Matlab library

Participants: Jonathan Grizou [correspondant], Iñaki Iturrate, Luis Montesano, Manuel Lopes, Pierre-Yves Oudeyer.

The Matlab software implements the algorithms described in [45]. It allows a robot to be instructed a new task by a human using communicative signals initially totally unknown to the robot. It is currently extended and improved in the context of EEG-based brain-machine interfaces (BMIs) [44].

It results in a BCI based control of sequential tasks with feedback signals that do not require any calibration process. As a by-product, the method provides an unsupervised way to train a decoder with the same performance of state-of-the-art supervised classifiers, while keeping the system operational and solving, with a lower performance during the first steps, the unknown task. The algorithm has been tested with online experiments (fig. 7), showing that the users were able to guide from scratch an agent to a desired position.



Figure 7. Results from the online BCI experiment for identifying the task. Evolution of the probability of the taught task for each subject and run

To improve the efficiency of the algorithm, we introduced a new planning method that uses the uncertainty in the signal-target estimation. This planner is inspired by exploration methods with exploration bonuses that allow guiding to reduce the uncertainty in an efficient way. We showed that trying to follow the best hypothesis

does not explore the space significantly to reduce uncertainty and thus identify the correct task. Only through an approach that plans how to reduce the uncertainty multiple steps ahead are we sure that the agent will reach states that can only be explained by the correct hypothesis.



Figure 8. Comparison between different exploration methods. Planning wrt. uncertainty in noth task and signal space is the most efficient method

5.3.5. *PROPRE: simulation of developmental concept formation using PYTHON*

Participant: Alexander Gepperth [correspondant].

This simulation software implements the algorithms described in [86], [83]. It is available online under the URL www.gepperth.net/downloads.html. The simulation is implemented in PYTHON for easy use, yet the time-critical core functions are written in C.

5.3.6. *pyStreamPlayer: synchronized replay of multiple sensor recordings and supplementary data*

Participant: Alexander Gepperth [correspondant].

This Python software is intended to facilitate the application of machine learning algorithms by avoiding to work directly with an embodied agent but instead with data recorded in such an agent. Assuming that non-synchronous data from multiple sensors (e.g., camera, Kinect, laser etc.) have been recorded according to a flexible format defined by the pyStreamPlayer architecture, pyStreamPlayer can replay these data while retaining the exact temporal relations between different sensor measurements. As long as the current task does not involve the generation of actions, this software allows to process sensor data as if it was coming from an agent which is usually considerably easier. At the same time, pyStreamPlayer allows to replay arbitrary supplementary information such as, e.g., object information, as if it was coming from a sensor. In this way, supervision information can be stored and accessed together with sensory measurements using a unified interface. pyStreamPlayer has been used to facilitate real-world object recognition tasks, and several of the major databases in this field (CalTech Pedestrian database, HRI RoadTraffic traffic objects database, CVC person database, KITTI traffic objects database) have been converted to the pyStreamPlayer format and now serve as a source of training and test data for learning algorithms.

pyStreamPlayer has been integrated into a ROS node as well, allowing the replay and transmission across networks of distributed processes.

5.3.7. *Multimodal: framework around the NMF algorithm for multimodal learning*

Participant: Olivier Mangin [correspondant].

The python code provides a minimum set of tools and associated libraries to reproduce the experiments on [53], together with the choreography datasets. The code, publicly available at <https://github.com/omangin/multimodal>, under the new BSD license, is primarily intended for reproduction of the multimodal learning experiment mentioned above. It is also expected that the public availability of the code encourages further experimentation by other scientists with data coming from other domains, thus increasing both the impact of the aforementioned publication and the knowledge on the algorithm behaviors. The nonnegative matrix factorization algorithm used in the experiments will also soon be included as a third party project to <http://scikit-learn.org>. Finally the code is currently being used by other members of the team and is expected to play an important role in further collaborations.

5.3.8. *Tools for curiosity-driven learning on a robotic arm*

Participants: Pierre Rouanet [correspondant], Clément Moulin-Frier.

This library is intended to provide tools for experimenting how curiosity model can facilitate learning of complex tasks such as manipulating objects. First, it provides high-level access to a robotic arm made of dynamixel motors: forward and inverse kinematics, demonstrations recording. Then it wraps the IMLE [77] library which we used for incremental and online learning of the sensorimotor mappings of the robot. Finally, it implements curiosity-driven learning based on the maximization of the learning progress. This modelling is based on recent works by Moulin-Frier and Oudeyer [54], [56], [55] proposing a probabilistic algorithmic architecture unifying various principles of developmental robotics such as motor babbling, goal babbling and curiosity-driven exploration. This architecture has already been successfully applied to model infant speech acquisition in the previously cited papers.

5.4. Software Platforms

5.4.1. *Meka robot platform enhancement and maintenance*

Participants: Antoine Hoarau [ADT Engineer Since Nov. 2012], Freck Stulp [Supervisor], David Filliat [Supervisor].

Autonomous human-centered robots, for instance robots that assist people with disabilities, must be able to physically manipulate their environment. There is therefore a strong interest within the FLOWERS team to apply the developmental approach to robotics in particular to the acquisition of sophisticated skills for manipulation and perception. ENSTA-ParisTech has recently acquired a Meka (cf. 9) humanoid robot dedicated to human-robot interaction, and which is perfectly fitted to this research. The goal of this project is to install state-of-the-art software architecture and libraries for perception and control on the Meka robot, so that this robot can be jointly used by FLOWERS and ENSTA. In particular, we want to provide the robot with an initial set of manipulation skills.

The goal is to develop a set of demos, which demonstrate the capabilities of the Meka, and provide a basis on which researchers can start their experiments.

The platform is evolving as the software (Ubuntu, ROS, our code) is constantly updated and requires some maintenance so less is needed for later. A few demos were added, as the hand shaking demo, in which the robot detects people via kinect and initiates a hand shake with facial expressions. This demo has been used to setup a bigger human robot interaction experiment, currently tested on subjects at Ensta (cf. 10). The stacking cups demo is more of a manipulation and vision demo : the robot detects its cups by their shape and color, and tries to make a tower with it (cf. 12). This demo has required to manually update the old pr2 tabletop object detector to the new ROS version, create a tool to semi-automatically calibrate the extrinsics parameters of the kinect, and different set of tools to catch objects from different angles (waiting for moveit to be fully integrated). Finally, we've seen that the robot itself also needs some maintenance; some components broke (a finger tendon), a welding got cold (in the arm) and a few cables experienced fatigue (led matrix and cameras) (cf. 11).

I've also given a talk at Humanoid 2013 at Atlanta and the University of Texas at Austin for my participation on [61].

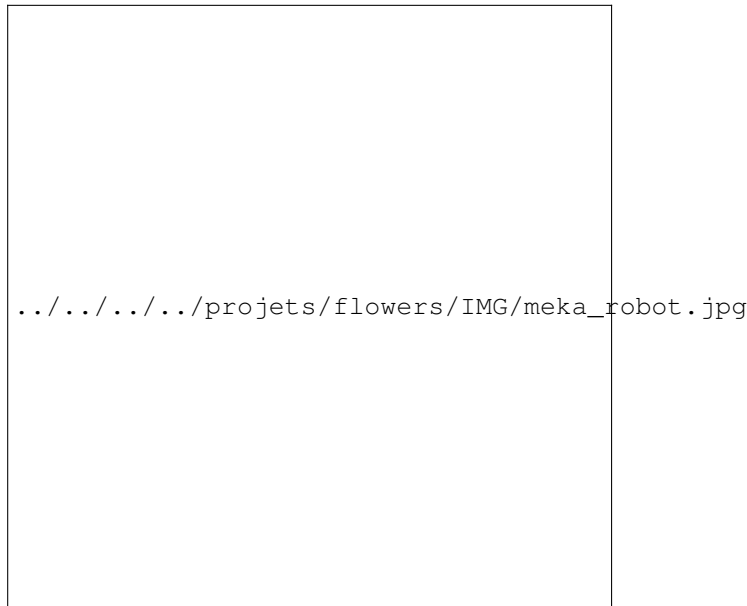


Figure 9. The Meka robot platform acquired by ENSTA ParisTech

5.4.2. Experiment platform for multiparameters simulations

Participants: Fabien Benureau, Paul Fudal.



Figure 10. Hand shake demo visualized on Rviz (ROS)



Figure 11. Maintenance is required on the robot



Figure 12. The stacking cup demo (video available at <http://cogrob.ensta-paristech.fr>)

Simulations in robotics have many shortcomings. At the same time, they offer high customizability, rapidity of deployment, absence of failure, consistency across time and scalability. In the context of the PhD work of Fabien Benureau, it was decided to investigate hypothesis first in simulation before moving to real hardware. In order to be able to test a high number of different hypothesis, we developed a software platform that would scale to the computing ressource available.

We designed simple continuous simulations around a of-the-shelf 2D physic engine and wrote a highly modular platform that would automatically deploy experiments on cluster environments, with proper handling of dependencies; our work investigate transfer learning, and some experiments's input data is dependent of the results of another.

So far, this platform and the university cluster has allowed to conduct thousands of simulations in parallel, totaling more than 10 years of simulation time. It has led us to present many diverse experiments in our published work [40], each repeated numerous times. It has allowed us to conduct a multi-parameter analysis on the setup, which led to new insights, which are being presented in a journal article to be submitted in the beginning of this year.

Because of its high modularity, this platform is proving to be highly flexible. We are currently adaptating it to a modified, cluster-ready, version of the V-REP simulator. Those simulations will serve to back ones on similar real-world hardware that are currently setup.

We have released the platform and the complete experiments code when we published the results of [40], allowing to reproduce the results of the paper, and will continue to do so with each published work.

5.4.3. PyPot

Participants: Pierre Rouanet [correspondant], Matthieu Lapeyre.

PyPot is a framework developed to make it easy and fast to control custom robots based on dynamixel motors. This framework provides different levels of abstraction corresponding to different types of use. More precisely, you can use PyPot to:

1. directly control robot's motors through a USB2serial device,
2. define the structure of your particular robot and control it through high-level commands,
3. define primitives and easily combine them to create complex behavior.

PyPot has been entirely written in Python to allow for fast development, easy deployment and quick scripting by non-necessary expert developers. It can also benefit from the scientific and machine learning libraries existing in Python. The serial communication is handled through the standard library and thus allows for rather high performance (10ms sensorimotor loop). It is cross-platform and has been tested on Linux, Windows and Mac OS.

PyPot is part of the Poppy project (<http://www.poppy-project.org>) and has been released under an open source license GPL V3. More details are available on PyPot website: <https://github.com/poppy-project/pypot>

5.5. Experimental Setups

5.5.1. Experimental Platform for User Study of Curiosity-driven Exploration

Participants: Pierre Rouanet [correspondant], Jonathan Grizou, Brice Miard, Julie Golliot.

This platform has been developed to investigate curiosity-driven behaviors and more precisely how humans explore new sensori-motor spaces. It consists in several simple games where users control a 2D/3D shape with the movements of their body. They have to discover the mapping between their movements and a shape displayed on the screen and learn how to make the controlled shape match the target one (fig 13).



Figure 13. A screenshot representing the game interface as seen by the user.

The software is entirely written in Python. It includes a Kinect wrapper allowing the access of 3D position of tracked skeleton joints. It provides a framework for creating new games based on the 2D drawing library (pygame). It also includes a web server used to display game instructions, cut-scene videos and questionnaire.

5.6. Visualization Tools

5.6.1. Zephyr - Realtime Visualization in JAVA

Participant: Thomas Degris [correspondant].

Zephyr is a software to visualize numeric variables and data structure in real time and at different time scale. Zephyr is practical because it requires only minimal changes in the code: it uses Java reflexivity to automatically detect variables in the code to monitor and data structure with an associated dedicated view. Zephyr can easily be extended with new plugins because it is based on the popular Eclipse Rich Client Platform. Consequently, Zephyr takes advantage of an already existing and fully operational Eclipse plugins for many of its functionalities. Finally, Zephyr is distributed with a Java python virtual machine named Jython and a lisp implementation named Clojure. An example of a Zephyr screen is shown in Figure 14 .

Zephyr was started in fall 2009 in the RLAI group at the university of Alberta (Canada) when Thomas Degris was a postdoc in this group. Zephyr is still actively used by RLAI. Users include Adam White, Joseph Modayil and Patrick Pilarski from the University of Alberta. Zephyr has been registered on the Eclipse marketplace since October 2011. Documentation about Zephyr is included on its website: <http://zephyrplugins.github.com>. Zephyr is licensed under the open source Eclipse Public License.

5.6.2. Experimental Setups for User Study of Alignment in Asymmetric Interactions

Participants: Jonathan Grizou [correspondant], Chloé Rozenbaum, Manuel Lopes, Katharina Rohlfing, Pierre-Yves Oudeyer.



Figure 14. Left: Zephyr showing the different steps of a video processing pipeline in real-time. Right: Zephyr showing different data structure and variables of a reinforcement learning agent at different time scale. A video is available at: <http://zephyrplugins.github.com>.

This platform has been developed to investigate alignment in asymmetric interactions. We consider a remote construction task, where one user (user A) knows what to build but do not have access to the construction site while its partner (user B) is at the site but do not know what to do. By constraining the communicative channel between the two partners, we study how, and if, they will agree on a similar set of signals to convey information and what type of information they tend to produce.

The experimental setup consist of box with button, a video recording system and two screens. User A can send signals to user B by pressing buttons (fig. 15). Signals are displayed on a screen (fig. 15) at user B side. User A is not aware of what is displayed on user B screen, neither user B is aware of the relation between button presses and screen events. The video of user B construction scene is streamed to a screen at user B side.

The task consist of bulding arbitrary construction (fig. 15) using colored toy bricks (fig. 15).

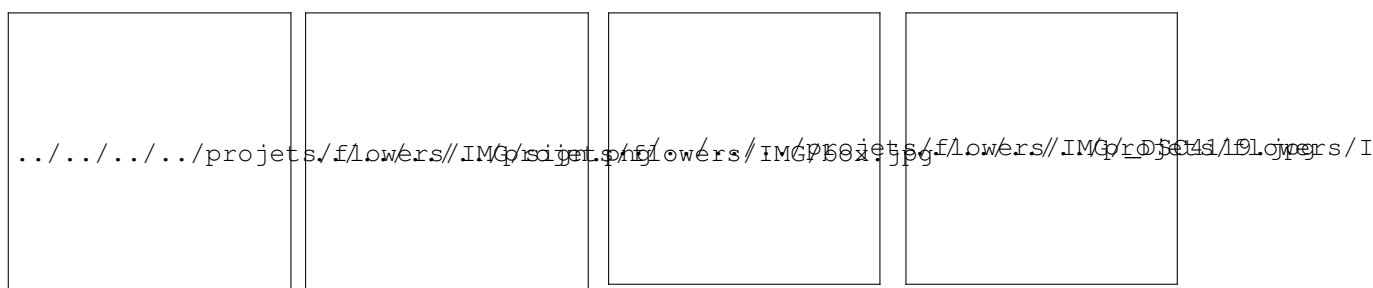


Figure 15. Three examples of sign displayed on the learner screen; The box and the button use as an interface for the teacher to communicate with the learner; Examples of construction presented to the teacher.

5.7. Hardware

5.7.1. Poppy Platform

Participants: Matthieu Lapeyre [correspondant], Pierre Rouanet, Jonathan Grizou, Pierre-Yves Oudeyer [supervisor].

5.7.1.1. Main goals :

No current platform (Nao [87], Darwin Op [88], Nimbro Op [126], HRP-2, ...) does offer both a adapted morphology in the sense of allowing physical interaction (safe, compliant, playful) and optimized for walking. So to explore these challenges we have decided to build a new bio-inspired humanoid robotic platform, called Poppy, which provides some of the software and hardware features needed to explore both social interaction and biped locomotion for personal robot. It presents the following main features to make it an interesting platform to study how the combination of morphology and social interaction can help the learning:

- Design inspired from the study of the anatomy of the human body and its bio-mechanic
- Dynamic and reactive: we try to keep the weight of the robot as low as possible (geometry of the pieces and smaller motors)
- Social interaction: screen for communication and permits physical interaction thanks to compliance
- Study of the morphology of the leg to improve the biped walking
- Practical platform: low cost, ease of use and easy to reproduce

5.7.1.2. Overview :

Poppy platform (Figure 16) is a humanoid, it is 84cm tall for 3 kg. It has a large sensor motors space including 25 dynamical motors (MX-28 and AX-12), force sensors under its feet and some extra sensors in the head: 2 HD-wide angle-cameras, stereo-micros and an inertial central unit (IMU 9DoF) plus a large LCD Screen (4 inch) for visual communication (e.g. emotions, instructions or debug). The mechanical parts were designed and optimized to be as light as possible while maintaining the necessary strength. For this, the choice of a lattice beam structure manufactured with 3Dprinting polyamide was used.

The poppy morphology is designed based on the actual human body. We have deeply studied the biomechanics of the human body and have extracted some interesting features for humanoid robotics. This inspiration is expressed in the whole structure (e.g. the limb proportions) and in particular in the trunk and legs.



Figure 16. a. Global view of the Poppy platform. b. Zoom on legs design

Poppy uses the bio-inspired trunk system introduced by Acroban [101]. These five motors allow it to reproduce the main changes brought by the human spine. This feature allows the integration of more natural and fluid motion while improving the user experience during physical interactions. In addition, the spine plays a fundamental role in bipedal walking and postural balance by actively participating in the balancing of the robot.

The legs were designed to increase the stability and agility of the robot during the biped walking by combining bio-inspired, semi-passive, lightweight and mechanical-computation features. We will now describe two examples of this approach:

The architecture of the hips and thighs of Poppy uses biomechanical principles existing in humans. The human femur is actually slightly bent at an angle of about 6 degrees. In addition, the implantation of the femoral head in the hip is on the side. This results in a reduction of the lateral hip movement needed to move the center of gravity from one foot to another and a decrease in the lateral falling speed. In the case of Poppy, the inclination of its thighs by an angle of 6 degrees causes a gain of performance of more than 30% for the two above mentioned points.

Another example is Poppy's feet. Poppy has the particularity of having small feet compared to standard humanoids. It has humanly proportioned feet (ie about 15% of its total size). It is also equipped with compliant toes joints (see Figure 17 .a). We believe that this feet involve two keys features to obtain a human-like and efficient walking gait. However, that raises problems regarding balance because the support polygon is reduced. We decided to add pressure sensors under each foot in order to get accurate feedback of the current state of the robot (see Figure 17 .b).



Figure 17. Poppy feet use actual children shoes combine with a compliant feet, toes (a.) and pressure sensors (b.)

5.7.1.3. Open source release :

To allow the distribution in the robotic community, we have decided to make Poppy an open platform. So the software and the hardware are open source. They have each a repository. The PyPot library is under GPLv3 license and is available on a bitbucket repository (<https://github.com/poppy-project/pypot>). The hardware, Solidworks files and STL needed to print the robot are available under a Creative Commons BY+SA+NC license on a private GrabCAD Workbench repository. People can request access to the hardware repository on the Poppy website (<http://www.poppy-project.org/open-platform/>).

The platform is currently under beta-testing meaning that we let the community grows little by little to ensure a good support of interesting projects.

For now, there are about 200 people on the GrabCAD project. They have access to all files needed to print the robot. There is also about 60 beta testers. They have access to a private section on the website with documentation and a forum for support.

Several of them are already doing a great work both by reporting bugs and managing to build the robot outside the lab. We are trying to work closely with them as they are a great source of feedback to improve the platform before a more wide distribution.

5.7.1.4. Impact in the community

Poppy has been release open source the 15/10/2013. The announcement has been done by Pierre Yves Oudeyer during the Lift 2013 conference. To prepare this event, the website and a overview video were made. The video, accessible here (<http://vimeo.com/76917854>) has reached about 40K views until now.

A part of the audience are technology-enthusiast people (about 50% on grabCAD), interested by the fact Poppy is 3D printable and so, highly customizable. More interesting, we received a large number of beta request from various applications domains (see 18) around the world (see 19).



Figure 18. The 60 selected beta testers profiles chart.

5.7.1.4.1. Education:

We receive a lot of request from Fablab around the world (France, USA, New-Zeland, South Africa,...). All have great educational projects for teaching robotics, mechanics and computer science to children. Some of them are close beta testers and we have weekly interactions. In the same topic, several schools, engineering schools and universities showed interest to use Poppy as experimental support. One of our close beta tester is a Bordeaux high school. They are planning to use Poppy as support for mechanics, computer sciences but also architecture or philosophy.

5.7.1.4.2. Art:

A current art project is under construction. A residence with an artist, a dancer and us will take place on the 24/02/2014 to the 05/03/2014. It comes to artistically invest humanoid robotics and thereby examine the relationship of the body to the digital world. The encounter between art and science generates potential new ideas for both disciplines who find themselves at the crossroads of questions relating to the gesture, movement and body. The Poppy Project is focused on morphological and motor aspects in a context human-robot interface. The look of an artist and the movements of a dancer are testing this interface with an unprecedented and direct manere.



Figure 19. Where are located the beta testers?

5.7.1.4.3. Research:

A large number of researchers showed interest in the platform. Most of them are interested to use it as a experimental tool. They want to address challenges such as balance and walking control, use of force controlled motors, explore human-robot interactions. On this last topic, the Bristol robotic lab target to use tele-operation to investigate what factors are important in terms of appearance and behaviour for credible and trustworthy interaction. For this purpose, they will develop functional hands for gesture and grasping task.

5.7.1.5. Next step:

We are currently working with beta testers on several improvements to make Poppy more accessible, more easy to use and more polyvalent. Two internship students will arrive on the project to work on the embedded electronic and on the feet design. We are targeting to release the final version during the summer 2014. Also we are thinking about the creation of a "Poppy pack" including all necessary components and tools to easily build the robot.

Another very important point is the community management. We are currently adding new collaborative tools on the Poppy website. These tools are tested with beta testers. The challenge is to offer the good tools to provide an efficient support to future users and to encourage people to contribute. This work is done in collaboration with Stephane Ribas (D2T inria Grenoble).

LAGADIC Project-Team

5. Software and Platforms

5.1. ViSP: a visual servoing and tracking software library

Participants: Fabien Spindler [correspondant], Aurélien Yol, Eric Marchand, François Chaumette.

Since 2005, we develop and release under the terms of the GPLv2 licence, ViSP, an open source library available from <http://team.inria.fr/lagadic/visp/visp.html>. It allows fast prototyping of visual tracking and visual servoing tasks. ViSP was designed to be independent with the hardware, to be simple to use, expandable and cross-platform.

ViSP allows to design vision-based tasks for eye-in-hand and eye-to-hand visual servoing that contains the most classical visual features that are used in practice. It involves a large set of elementary positioning tasks with respect to various visual features (points, segments, straight lines, circles, spheres, cylinders, image moments, pose,...) that can be combined together, and image processing algorithms that allows tracking of visual cues (dots, segments, ellipses,...) or 3D model-based tracking of known objects. Simulation capabilities are also available. ViSP and its full functionalities are presented in Fig. 1 and described in [6].

This year, we continued our efforts to improve the software by increasing the compatibility with exotic platforms, fixing issues, and by introducing an hybrid scheme in the model-based tracker to take advantage of texture. We also improved the documentation by providing tutorials covering the main capabilities of the software. Two releases were produced, one in February downloaded 1000 times and the other in July downloaded 730 times. With the help of the community, the last release was also packaged for Ubuntu 13.10. A new template tracker developed during A. Dame's Ph.D. was recently introduced and will be available in the next release.

Concerning ROS community, all the existing packages in "vision_visp" ROS stack (see http://www.ros.org/wiki/vision_visp) were updated and ported to catkin build system. To ease ViSP usage in the ROS framework, the last release was packaged for ROS.

ViSP is used in research labs in France, USA, Japan, Korea, India, China, Lebanon, Italy, Spain, Portugal, Hungary, Canada. For instance, it is used as a support in graduate courses at IFMA Clermont-Ferrand, University of Picardie in Amiens, Télécom Physique in Strasbourg and ESIR in Rennes.

5.2. DESlam software

Participant: Patrick Rives [correspondant].

The DESlam (Dense Egocentric Slam) software developed in collaboration with Andrew Comport from I3S in Sophia Antipolis was registered to the APP ("Agence de Protection des Programmes") (IDDN.FR.001.320001.000.S.P.2012.000.21000). This software proposes a full and self content solution to the dense Slam problem. Based on a generic RGB-D representation valid for various type of sensors (stereovision, multi-cameras, RGB-D sensors...), it provides a 3D textured representation of complex large indoors or outdoors environments and it allows to localize in real time (45Hz) a robot or a person carrying out a mobile camera.

5.3. Robot vision platforms

Participant: Fabien Spindler [correspondant].

We exploit two industrial robotic systems built by Afma Robots in the nineties to validate our researches in visual servoing and active vision. The first one is a Gantry robot with six degrees of freedom, the other one is a cylindrical robot with four degrees of freedom (see Fig. 2). These robots are equipped with cameras. The Gantry robot allows also to embed grippers on its end-effector.

../../../../projets/lagadic/IMG/visp1.png

../../../../projets/lagadic/IMG/visp2.png



Figure 2. Lagadic robotics platforms for vision-based manipulation

Three papers published by Lagadic in 2013 enclose results validated on this platform.

5.4. Medical robotics platforms

Participants: Fabien Spindler [correspondant], Alexandre Krupa.

This testbed is of primary interest for researches and experiments concerning ultrasound visual servoing applied to positioning or tracking tasks described in Section 6.4 .

This platform is composed by two Adept Viper six degrees of freedom arms (see Fig. 3). Ultrasound probes connected either to a SonoSite 180 Plus or an Ultrasonix SonixTouch imaging system can be mounted on a force torque sensor attached to each robot end-effector.

We started experimentation to validate needle detection and tracking under ultrasound imaging (see Section 6.4.1).

This year, two papers enclose experimental results obtained with this platform.



Figure 3. Lagadic medical robotics platforms. On the right Viper S850 robot arm equipped with a SonixTouch 3D ultrasound probe. On the left Viper S650 equipped with a tool changer that allows to attach a classical camera.

5.5. Mobile robotics platforms

Participants: Fabien Spindler [correspondant], Erwan Demairy, Marie Babel, Patrick Rives.

5.5.1. Indoors mobile robots

For fast prototyping of algorithms in perception, control and autonomous navigation, the team uses Hannibal in Sophia Antipolis, a cart-like platform built by Neobotix (see Fig. 4 .a), and a Pioneer 3DX from Adept in Rennes (see Fig. 4 .b) as well as a Robotino from Festo. These platforms are equipped with various sensors needed for Slam purposes, autonomous navigation and sensor-based control.

Moreover, to validate the researches in personally assisted living topic (see 6.3.4), we bought in Rennes a six wheel electric wheelchair from Penny and Giles Drives Technology (see Fig. 4 .c). The control of the wheelchair is performed using a plug and play system between the joystick and the low level control of the wheelchair. Such a system let us acquire the user intention through the joystick position and control the wheelchair by applying corrections to its motion. The wheelchair has been fitted with cameras to perform the required servoing for assisting handicapped people. Moreover, to ensure the direct security of the user, seven infrared proximity sensors have been installed all around the wheelchair.

Note that three papers exploiting the indoors mobile robots were published this year.

5.5.2. Outdoors mobile robots

The team exploits also Cycab urban electrical cars (see Figs. 4 .d and 4 .e). Two vehicles in Sophia Antipolis and one in Rennes are instrumented with cameras and range finders to validate researches in the domain of intelligent urban vehicle. Cycabs were used as experimental testbeds in several national projects.

Three papers published by Lagadic in 2013 enclose experimental results obtained with these outdoors mobile robots.

5.5.3. Technological Development Action (ADT) P2N

The ADT P2N aims at sharing existing and in development codes between the Lagadic and E-Motion teams in the field of autonomous navigation of indoors robots. These codes are also used in the platforms involved in the large-scale initiative action PAL (Personnally Assisted Living, see Section 8.2.6). This year, the most notable activities for this ADT have been to:

- adapt a navigation module developed by E-Motion to the mobile platform used at Sophia-Antipolis;
- make the SLAM module developed by Lagadic usable by the E-Motion navigation module;
- port the code on the wheelchairs used in PAL;
- develop the core architecture running under ROS supporting the different sensors and platforms available in Sophia-Antipolis.

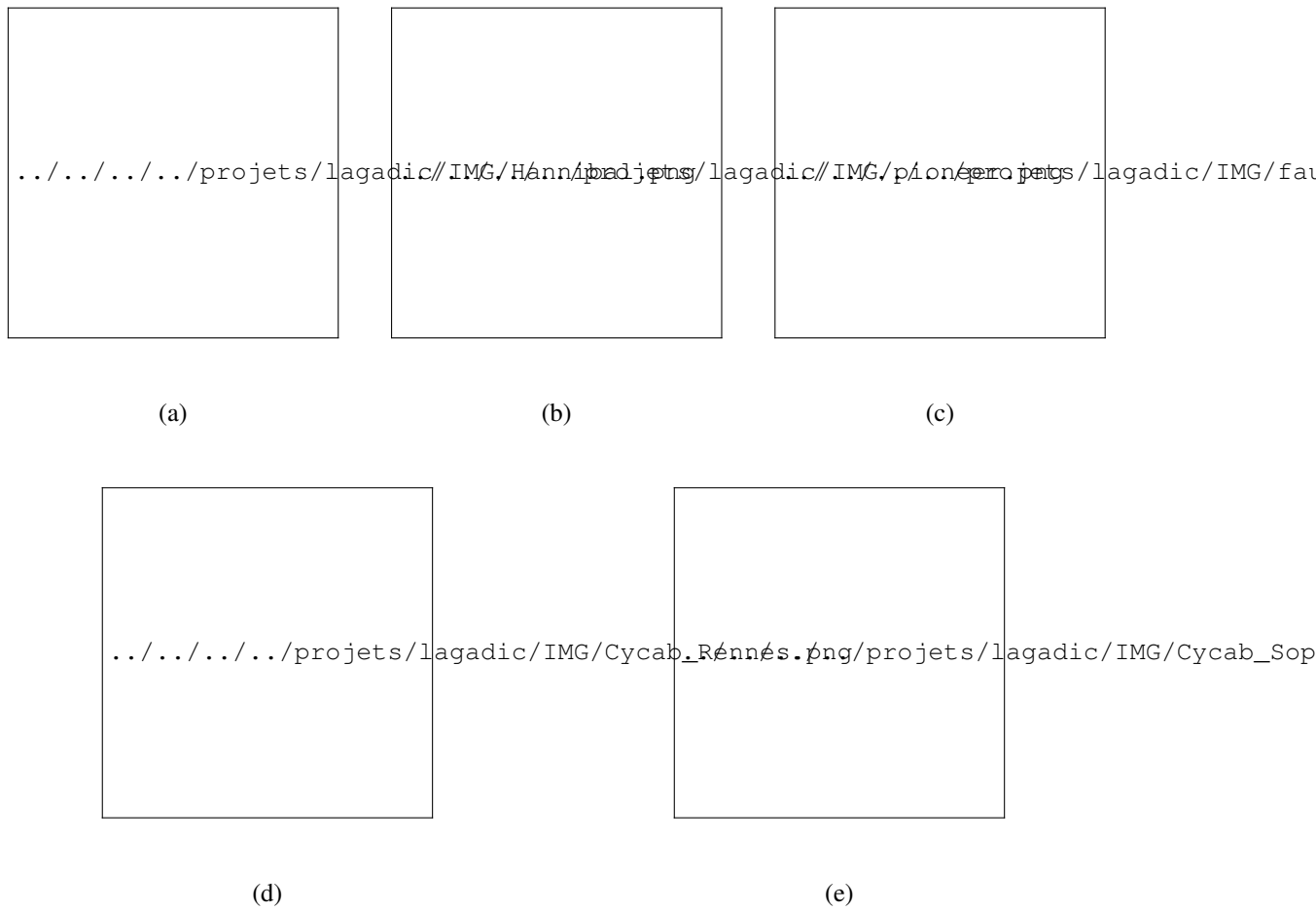


Figure 4. a) Hannibal platform, b) Pioneer P3-DX robot, c) six wheel electric wheelchair, d) Cycab available in Rennes, e) one of the Cycabs available in Sophia Antipolis.

AYIN Team

5. Software and Platforms

5.1. Deposits

- The software SAMD (Semi-Automatic Melanoma Detection) V1.0 was deposited with the APP in December 2013. It has been tested on public databases.
- The software SAAD (Semi-Automatic Acne Detection) V1.0 was deposited with the APP in December 2013. It has been tested on public databases as well as on data sets provided by CHU Nice and Galderma.

LEAR Project-Team

5. Software and Platforms

5.1. Large-scale image classification

Participants: Matthijs Douze [correspondant], Zaid Harchaoui, Florent Perronnin [XRCE], Cordelia Schmid.

JSGD is the implementation of a Stochastic Gradient Descent algorithm used to train linear multiclass classifiers. It is biased towards large classification problems (many classes, many examples, high-dimensional data). It can be used on the ImageNet large scale classification challenge. It uses several optimization techniques, both algorithmic (scale factors to spare vector multiplications, vector compression with product quantizers) and technical (vector operations, multithreading, improved cache locality). It has Python and Matlab interfaces. It is distributed under a Cecill licence. Project page: <http://lear.inrialpes.fr/src/jsgd>.

5.2. Fisher vector image representation

Participants: Matthijs Douze [correspondant], Hervé Jégou [TEXMEX Team Inria Rennes], Cordelia Schmid.

We developed a package that computes Fisher vectors on sparse or dense local SIFT features. The dense feature extraction was optimized, so that they can be computed in real time on video data. The implementation was used for several publications and in our submission to the Trecvid 2013 MED task. We provide a binary version of the local descriptor implementation, and the Fisher implementation is integrated in the Yael library, with Python and Matlab interface, see http://lear.inrialpes.fr/src/inria_fisher.

5.3. Video descriptors

Participants: Clement Leray, Dan Oneata, Cordelia Schmid [correspondant], Heng Wang, Jakob Verbeek.

We have developed and made on-line available software for video description based on dense trajectories and motion boundary histograms. The trajectories capture the local motion information of the video. A state-of-the-art optical flow algorithm enables a robust and efficient extraction of the dense trajectories. Descriptors are aligned with the trajectories and based on motion boundary histograms (MBH) which are robust to camera motion. This year we have further developed this software to increase its robustness and scalability to large datasets. Most importantly, we have added a robust background stabilization technique, which allows to remove camera motion. This has shown to significantly improve the performance. Furthermore, we have improved the efficiency of the approach. For example, we avoid writing the raw MBH descriptors to disk, but rather aggregate them directly into a signature for the complete video using Fisher vectors. This allowed us to use these descriptors on the 4,000 hour video dataset of the TrecVid 2013 MED task as well as on the 3500 hours of AXES broadcast videos.

5.4. SParse Modeling Software (SPAMS)

Participants: Julien Mairal [correspondant], Jean-Paul Chieze [WILLOW Project-Team], Jean Ponce [WILLOW Project-Team], Francis Bach [SIERRA Project-Team].

SPAMS v2.4 was released as open-source software in December 2013 (v1.0 was released in September 2009). It is an optimization toolbox implementing algorithms to address various machine learning and signal processing problems involving

- Dictionary learning and matrix factorization (NMF, sparse PCA, ...);
- Solving medium-scale sparse decomposition problems with LARS, coordinate descent, OMP, SOMP, proximal methods;
- Solving large-scale sparse estimation problems with stochastic optimization;
- Solving structured sparse decomposition problems (sparse group lasso, tree-structured regularization, structured sparsity with overlapping groups,...).

The software and its documentation are available at <http://spams-devel.gforge.inria.fr/>.

This year, we added new functionalities to the toolbox. A graphical tool for visualizing dictionaries was developed by Jean-Paul Chieze, and stochastic optimization tools corresponding to the papers [24], [23] were added for dealing with large-scale sparse estimation problems.

5.5. FlipFlop: Fast Lasso-based Isoform Prediction as a Flow Problem

Participants: Elsa Bernard [Institut Curie, Ecoles des Mines-ParisTech], Laurent Jacob [CNRS, LBBE Laboratory], Julien Mairal [correspondant], Jean-Philippe Vert [Institut Curie, Ecoles des Mines-ParisTech].

FlipFlop is an open-source software, implementing a fast method for de novo transcript discovery and abundance estimation from RNA-Seq data [36]. It differs from classical approaches such as Cufflinks by simultaneously performing the identification and quantitation tasks using a penalized maximum likelihood approach, which leads to improved precision/recall. Other softwares taking this approach have an exponential complexity in the number of exons of a gene. We use a novel algorithm based on network flow formalism, which gives us a polynomial runtime. In practice, FlipFlop was shown to outperform penalized maximum likelihood based softwares in terms of speed and to perform transcript discovery in less than 1/2 second for large genes.

FlipFlop 1.0.0 is a user friendly bioconductor R package. It is freely available on the Bioconductor website under a GPL licence: <http://bioconductor.org/packages/release/bioc/html/flipflop.html>.

5.6. DeepFlow

Participants: Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, Cordelia Schmid.

We developed a package for the "deep flow" algorithm [31]. "Deep flow" combines a standard variational framework with a our new matching algorithm "deep matching". The code for "deep matching" is in python and the code for "deep flow" in C. Both of them are available on-line at <http://lear.inrialpes.fr/src/deepmatching>. Note that the run time is a few seconds per images pair, which is less than for most other methods.

5.7. Object category localization

Participants: Ramazan Cinbis, Matthijs Douze, Cordelia Schmid, Jakob Verbeek.

We developed an object category localization system based on a Fisher vector representation over densely extracted local SIFT descriptors [18]. To improve the robustness with respect to background clutter in the detection windows we developed an approximate object segmentation method that is used to weigh the contribution of local SIFT descriptors. Our system achieves state-of-the-art localization performance as measured on the PASCAL VOC 2007 and 2010 datasets. The system is developed in both C, python, and Matlab. The system will be released in early 2014.

MAGRIT Project-Team

5. Software and Platforms

5.1. Software and Platforms

Our software efforts are integrated in a library called RALib which contains our research development on image processing, registration (2D and 3D) and visualization. This library is licensed by the APP (French agency for software protection).

The visualization module is called QGLSG: it enables the visualization of images, 2D and 3D objects under a consistent perspective projection. It is based on Qt ¹ and OpenScenegraph ² libraries. The QGLSG library integrates innovative features such as online camera distortion correction, and invisible objects that can be incorporated in a scene so that virtual objects can cast shadows on real objects, and occlusion between virtual and real objects are easier to handle. The library was also ported to Mac OS and Windows and a full doxygen documentation was written.

¹<http://qt.digia.com>

²<http://www.openscenegraph.org/projects/osg>

MORPHEO Team

5. Software and Platforms

5.1. Platforms

5.1.1. The Grimage platform

The Grimage platform is an experimental multi-camera platform dedicated to spatio-temporal modeling including immersive and interactive applications. It hosts a multiple-camera system connected to a PC cluster, as well as visualization facilities including head mounted displays. This platform is shared by several research groups, most prominently Moais, Morpheo and Perception. In particular, Grimage allows challenging real-time immersive applications based on computer vision and interactions between real and virtual objects, Figure 1 . Note that the Grimage platform will be replaced by the Kinovis platform that will exhibit a larger acquisition space and better acquisition facilities.



Figure 1. Platform: the Grimage acquisition.

5.1.2. Kinovis

Kinovis (<http://kinovis.inrialpes.fr/>) is a new multi-camera acquisition project that was selected within the call for proposals "Equipements d'Excellence" of the program "Investissement d'Avenir" funded by the French government. The project involves 2 institutes: the Inria Grenoble Rhône-Alpes, the université Joseph Fourier and 4 laboratories: the LJK(laboratoire Jean Kuntzmann - applied mathematics), the LIG (laboratoire d'informatique de Grenoble - Computer Science), the Gipsa lab (Signal, Speech and Image processing) and the LADAF (Grenoble Hospitals - Anatomy). The Kinovis environment will be composed of 2 complementary platforms. A first platform located at the Inria Grenoble will have a 10mx10m acquisition surface and will be equipped with 60 cameras. It is the evolution of the Grimage platform previously described towards the production of better models of more complex dynamic scenes. A second platforms located at Grenoble Hospitals, within the LADAF anatomy laboratory, will be equipped with both color and X-ray cameras to enable combined analysis of internal and external shape structures, typically skeleton and bodies of animals. Installation works of both platforms started in 2013 and should be finished in 2014. Members of Morpheo are highly involved in this project. Edmond Boyer is coordinating this project and Lionel Reveret is in charge of the LADAF platform.

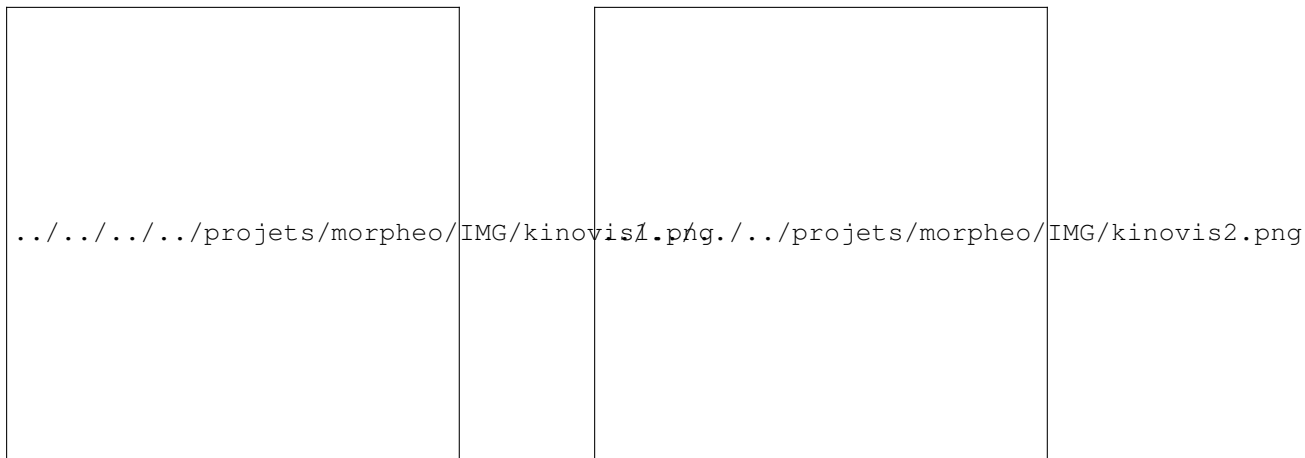


Figure 2. Kinovis platforms: on the left the Inria platform; on the right Grenoble Hospital platform.

5.1.3. Multicamera platform for video analysis of mice behavior

This project is a follow-up of the experimental set-up developed for a CNES project with Mathieu Beraneck from the CESeM laboratory (centre for the study of sensorimotor control, CNRS UMR 8194) at the Paris-Descartes University. The goal of this project was to analyze the 3D body postures of mice with various vestibular deficiencies in low gravity condition (3D posturography) during a parabolic flight campaign. The set-up has been now adapted for new experiments on motor-control disorders for other mice models. This experimental platform is currently under development for a broader deployment for high throughput phenotyping with the technology transfer project ETHOMICE. This project involves a close relationship with the CESeM laboratory and the European Mouse Clinical Institute in Strasbourg (Institut Clinique de la Souris, ICS).

5.2. Software packages

5.2.1. LucyViewer

Lucy Viewer http://4drepository.inrialpes.fr/lucy_viewer/ is an interactive viewing software for 4D models, i.e, dynamic three-dimensional scenes that evolve over time. Each 4D model is a sequence of meshes with associated texture information, in terms of images captured from multiple cameras at each frame. Such data is available from various websites over the world including the 4D repository website hosted by Inria Grenoble <http://4drepository.inrialpes.fr/>. The software was developed in the context of the European project iGance, it is available as an open source software under the GNU LGPL Licence.

5.2.2. Ethomice

Ethomice <http://morpheo.inrialpes.fr/people/reveret/ethomice/> is a motion analysis software to characterize motor behavior of small vertebrates such as mice or rats. From a multiple views video input, a biomechanical model of the skeleton is registered. Study on animal model is the first important step in Biology and Clinical research. In this context, the analysis of the neuro-motor behaviour is a frequent cue to test the effect of a gene or a drug. Ethomice is a platform for simulation and analysis of the small laboratory animal, such as rat or mouse. This platform links the internal skeletal structure with 3D measurements of the external appearance of the animal under study. From a stream of multiple views video, the platform aims at delivering a three dimensional analysis of the body posture and the behaviour of the animal. The software was developed by Lionel Reveret and Estelle Duveau. An official APP repository has been issued this year.



Figure 3. Ethomice: Experimental platform for video analysis of mice behavior.

5.3. Databases

5.3.1. 4D repository (<http://4drepository.inrialpes.fr/>)

This website hosts dynamic mesh sequences reconstructed from images captured using a multi-camera set up. Such mesh-sequences offer a new promising vision of virtual reality, by capturing real actors and their interactions. The texture information is trivially mapped to the reconstructed geometry, by back-projecting from the images. These sequences can be seen from arbitrary viewing angles as the user navigates in 4D (3D geometry + time) . Different sequences of human / non-human interaction can be browsed and downloaded from the data section. A software to visualize and navigate these sequences is also available for download.

PERCEPTION Team

5. Software and Platforms

5.1. Mixed camera platform

We started to develop a multiple camera platform composed of both high-definition color cameras and low-resolution depth cameras. This platform combines the advantages of the two camera types. On one side, depth (time-of-flight) cameras provide relatively accurate 3D scene information. On the other side, color cameras provide information allowing for high-quality rendering. The software package developed during the year 2011 contains the calibration of TOF cameras, alignment between TOF and color cameras, and image-based rendering. These software developments were performed in collaboration with the Samsung Advanced Institute of Technology. The multi-camera platform and the basic software modules are products of 4D Views Solutions SAS, a start-up company issued from the PERCEPTION group.



Figure 1. The mixed multi-camera system composed of four TOF-stereo sensor units.

5.2. Audiovisual robot heads

We have developed two audiovisual (AV) robot heads: the POPEYE head and the NAO stereo head. Both are equipped with a binocular vision system and with four microphones. The software modules comprise

stereo matching and reconstruction, sound-source localization and audio-visual fusion. POPEYE has been developed within the European project POP (<https://team.inria.fr/perception/pop/>) in collaboration with the project-team MISTIS and with two other POP partners: the Speech and Hearing group of the University of Sheffield and the Institute for Systems and Robotics of the University of Coimbra. The NAO stereo head was developed under the European project HUMAVIPS (<http://humavips.inrialpes.fr>) in collaboration with Aldebaran Robotics (which manufactures the humanoid robot NAO) and with the University of Bielefeld, the Czech Technical Institute, and IDIAP. The software modules that we develop are compatible with both these robot heads.

For more information on POPEYE and on NAO please visit <https://team.inria.fr/perception/popeye/> and <https://team.inria.fr/perception/nao/>.



Figure 2. Left: The consumer humanoid robot NAO is equipped with a binocular-binaural head specially designed for human-humanoid interaction; Right: The binocular-binaural robot head POPEYE equipped with a four degrees of freedom stereo camera pair and with a dummy head.

Prima Project-Team

4. Software and Platforms

4.1. OMiSCID Middleware for Distributed Multimodal Perception

Participants: Rémi Barraquand, Amaury Nègre, Patrick Reignier, Dominique Vaufreydaz [correspondant].

Middleware, Distributed perceptual systems

OMiSCID is lightweight middleware for dynamic integration of perceptual services in interactive environments. This middleware abstracts network communications and provides service introspection and discovery using DNS-SD (*DNS-based Service Discovery* [31]). Services can declare simplex or duplex communication channels and variables. The middleware supports the low-latency, high-bandwidth communications required in interactive perceptual applications. It is designed to allow independently developed perceptual components to be integrated to construct user services. Thus our system has been designed to be cross-language, cross-platform, and easy to learn. It provides low latency communications suitable for audio and visual perception for interactive services.

OMiSCID has been designed to be easy to learn in order to stimulate software reuse in research teams and is revealing to have a high adoption rate. To maximize this adoption and have it usable in projects involving external partners, the OMiSCID middleware has been released under an open source licence. To maximize its target audience, OMiSCID is available from a wide variety of programming languages: C++, Java, Python and Matlab. A website containing informations and documentations about OMiSCID has been set up to improve the visibility and promote the use of this middleware.

The OMiSCID graphical user interface (GUI) is an extensible graphical application that facilitates analysis and debugging of service oriented applications. The core functionality of this GUI is to list running services, their communication channels and their variables. This GUI is highly extensible and many modules (i.e. plugins) have been created by different members of the team: figure 2 shows an example of some of these modules. OMiSCID GUI is based on the Netbeans platform and thus inherits from its dynamic installation and update of modules.

4.2. Pal-Gate

Participants: Rémi Barraquand, Amaury Nègre, Dominique Vaufreydaz [correspondant].

Middleware, Distributed perceptual systems, Robotic Operating System (ROS), IPL PAL

A part of our efforts in the PAL project has been put toward developing a solution that would ease the integration of our multi-partners' software components. We refer to this solution as PALGate.

The design of PALGate results from the obvious observation that, within the PAL project, each partner must be considered as an ecosystem characterized, among other things, by 1) its software culture e.g. its curiosity and knowledge about software concepts, software architectures and design patterns, programatic languages, etc.; 2) its resources, e.g. its manpower, its possession or not of an experimental platform; 3) its competences and fields of research and expertise; 4) its habits e.g. its uses of a particular programming language, (c/c++, Java, Python) and computing platforms (OSx, Linux, Windows, Android, etc.), its adoption or not of a dedicated technology to interconnect software components (OSGi, OMiSCID, MPI, PVM, etc.); and 5) its particular needs and constraints e.g. requirement of a hard real-time system, mobility, etc.

For it to be widely accepted, PALGate is therefore designed to be ecologic and pragmatic. Ecologic in the sense that the solution does not perturb the ecology of each ecosystem¹, pragmatic in the sense that setting up this solution did not require an heavy development effort, also because it was targetted to PAL and is taking as much as possible advantage of existing solutions.

¹namely, if a partner is used to Java and OSGi, deploying PALGate will not affect this in any way nor engender an heavy effort to interface it.



Figure 2. OMiSCID GUI showing a list of running services and some modules for service interconnections, variable plotting, live video stream display and variable control

For developing PALGate we introduced a novel concept: software gate. Unlike software components/services which can be instantiated, a software gate is only a concept, it is defined as an ecologic and hermetic interface between different ecosystems. A software gate is characterized by the subset of functionalities it exposes to other gates, where the functionalities it exposes are provided by the software components/services of its belonging ecosystem. A software gate is hermetic in the sense that only a selected subset of functionalities of an ecosystem are exposed but also because it propagates only filtered information exposed by other gates into its ecosystem. The last characteristic of a software gate is that it makes explicit to other gates the communication mechanisms it uses.

While a software gate is only conceptual, PALGate is an implementation of a gate oriented middleware. PALGate uses ROS to support the basic communication between gates. Within PALGate, each ecosystem is associated to only one software gate. Practically, PALGate 1) is a ROS stack containing gates definition 2) is a set of conventions (e.g. stack organization, package/node/topic/service names, namespaces, etc.) 3) it provides dedicated tools to ease the integration and its usage by partners. A software gate in PALGate is a ROS package containing definition of ROS types (i.e. msgs and srvs types), but also exposed ROS communication channels (i.e. topics and RPCs).

With this architecture each partner has to provide PALGate with a package containing the definition of its gate. Then in order a) to expose functionalities out of their ecosystem and b) to propagate information into their ecosystem, each partner must create ROS nodes. These ROS nodes let each partner interface their ecosystem through ROS topics and ROS services without having to change anything about their architecture. For instance if a partner is using Java and OSGi, it can create nodes in ROS Java that will expose/register functionalities through ROS services, publish/subscribe information using ROS topics.

4.3. EmoPRAMAD

Participants: Claudine Combe, Dominique Vaufreydaz [correspondant].

Affective computing,

Within the Pramad project, we want to offer a full affective loop between the companion robot and the elderly people at home. This affective loop is necessary within the context of everyday interaction of elderly and the companion robot. A part of this loop is to make the robot express emotions in response to the emotional state of the user. To do that, we need to test our working hypothesis about the visual representation of emotions with the 3D face of robot. EmoPRAMAD is an evaluation tool designed to conduct comparative studies between human faces and the 3D faces expressing a defined set of emotions.

The evaluation conducted though EmoPRAMAD concerns both unimodal (facial only) and bimodal conditions (facial/sound). The emotions set is composed of 4 basic emotions (joy, fear, anger, sadness) and a neutral state. While experimenting, the software collects several parameters in order to evaluate more than correctness of the answers: time to respond, length of mouse moves, etc. Experimentation is still in progress at Inria in Grenoble, University Pierre and Marie Currie and Broca Hospital in Paris. A set of 235 participants from 14 to 88 years old was already recorded.

4.4. Detection and Tracking of Pedestrians in INRETS Intelligent Urban Spaces Platform

Participants: Claudine Combe, James Crowley [correspondant], Lukas Rummelhard.

Visual detection and tracking of pedestrians, Intelligent Urban Space

The project ANR-07-TSFA-009-01 CIPEBUS ("Carrefour Intelligent - Pole d'Echange - Bus) has been proposed by INRETS-IFSTTAR, in collaboration with Inria, Citilog, Fareco, and the city of Versailles. The Objective of the CIPEBUS project is to develop an experimental platform for observing activity in a network of urban streets in order to experiment with techniques for optimizing circulation by context aware control of traffic lights.

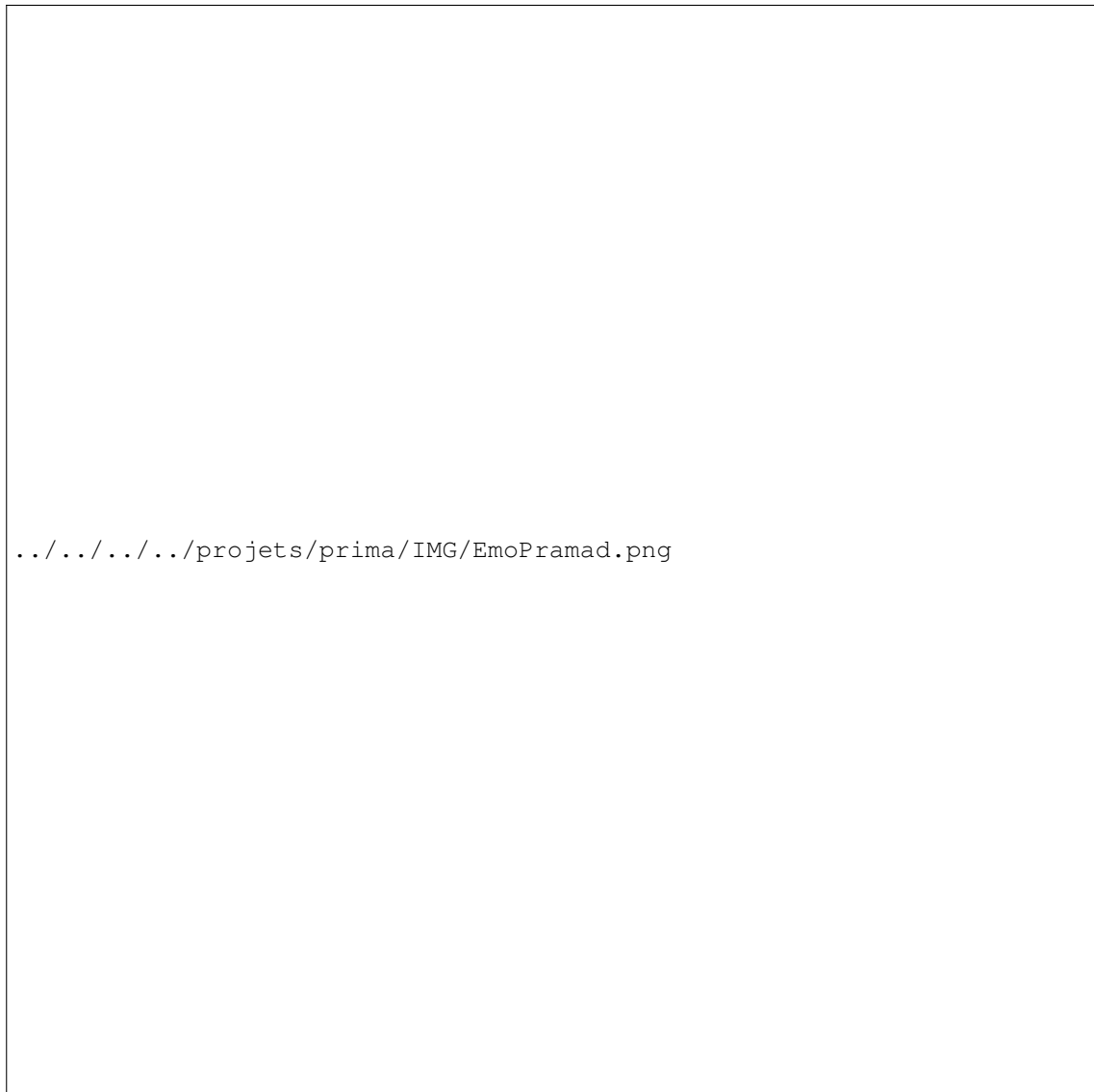


Figure 3. EmoPRAMAD interfaces with a human face and a 3D face from our virtual agent.

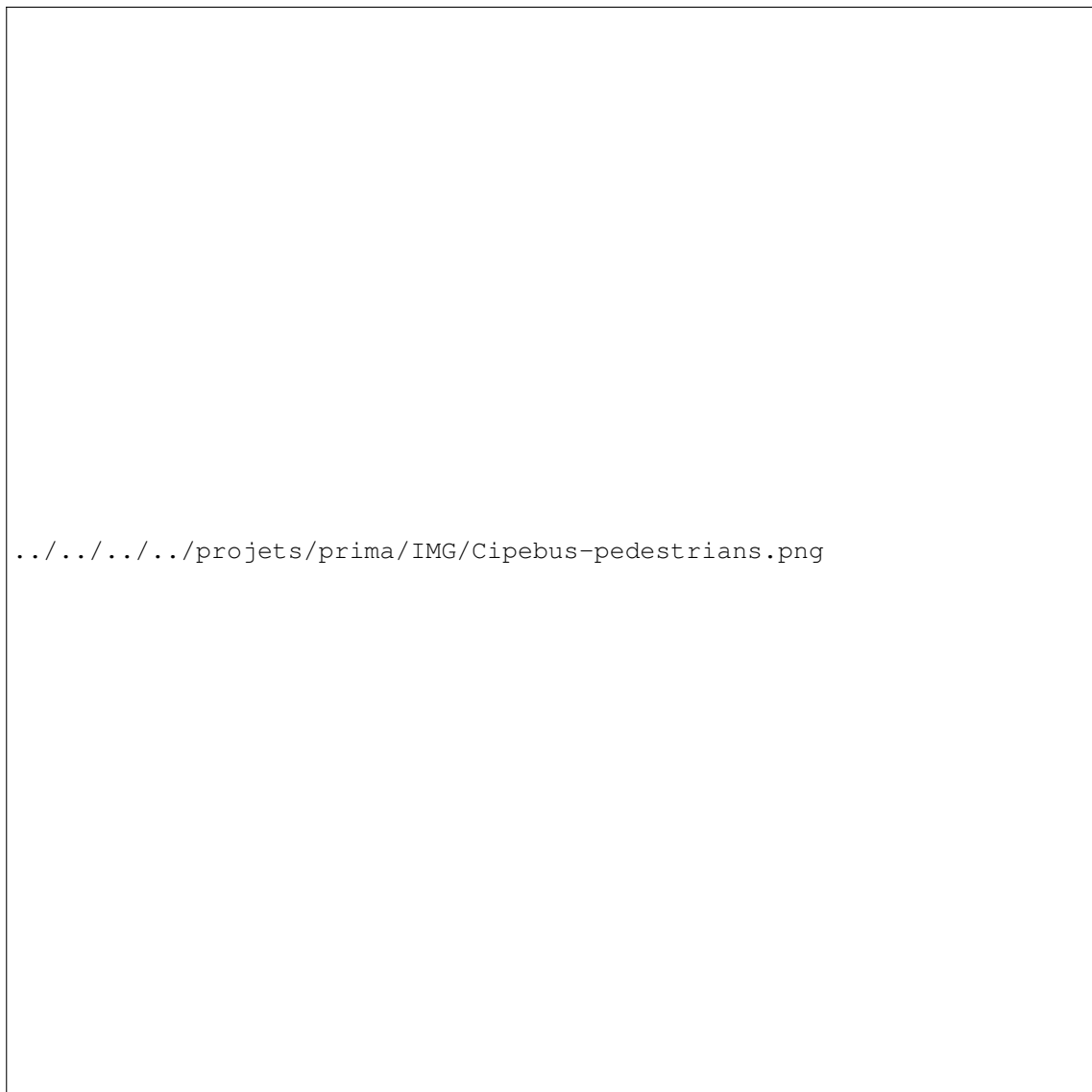


Figure 4. Cipebus: pedestrian tracking system.

Within CipeBus, Inria has developed a real time multi-camera computer vision system to detect and track people using a network of surveillance cameras. The CipeBus combines real time pedestrian detection with 2D and 3D Bayesian tracking to record the current position and trajectory of pedestrians in an urban environment under natural view conditions. The system extends the sliding window approach to use a half-octave Gaussian Pyramid to explore hypotheses of pedestrians at different positions and scales. A cascade classifier is used to determine the probability that a pedestrian can be found at a particular position and scale. Detected pedestrians are then tracked using a particle filter.

The resulting software system has been installed and tested at the INRETS CipeBus platform and is currently used for experiments in controlling the traffic lights to optimize the flow of pedestrians and public transportation while minimizing the delay imposed on private automobiles.

4.5. Multisensor observation of human activity for integrated energy and comfort management

Participants: Claudine Combe, James Crowley [correspondant], Lucas Nacsá, Amaury Nègre, Lukas Rummelhard.

multimodal tracking of human activity

As part of Inria's contribution of ICTLabs Action TSES - Smart Energy Systems, we have constructed a system that integrates information from multiple environmental sensor to detect and track people in indoor environments. This system, constructed as part of activity 11831 Open SES Experience Labs for Prosumers and New Services, has been released to ICTLabs partners in June 2012. It has also been used for construction of a smart spaces testbed at Schneider Electric.

This software, named MultiSensor activity tracker, integrates information from multiple environmental sensors to keep track of the location and activity of people in a smart environment. This model is designed to be used by a home energy broker that would work in conjunction with a smart grid to manage the energy consumption of home appliances, balancing the needs of inhabitants with opportunities for savings offered by electricity rates. This database will also be used for by advisor services that will offer advice to inhabitants on the consequences to energy consumption and energy cost that could potentially result from changes to lifestyle or home energy use.

Work in this task draws from earlier result from a number of development projects at Inria. In the ANR Casper project Inria created Bayesian tracking system for human activity using a voxel based occupancy grid. Within the INRA ADT PAL project, Inria is creating methods for plug and play installation of visual and acoustic sensors for tracking human activity within indoor environments.

While a voxel based Bayesian tracker has served well for a number of applications, a number of limitations have been observed. For example, under certain circumstances, the sensor data can provide contradictory or ambiguous data about the location and activities of people. Resolving such cases required the Bayesian tracker to choose between a numbers of competing hypotheses, potentially resulting in errors. Several members of our group have argued that an alternative integration approach based on the use of a Particle filter would solve these problems and provide a more reliable tracking system. This task has been undertaken to evaluate this hypothesis. The system configured and optimized for detecting and tracking people within rooms using multiple calibrated cameras. The system currently uses corner mounted cartesian cameras, ceiling mounted cameras with wide angle lenses and panoramic cameras placed on tables. Cameras may be connected and disconnected while the component is running, but they must be pre-calibrated to a common room reference frame. We are currently experimenting with techniques for Bayesian estimation of camera parameters for auto-calibration. Cameras may be connected dynamically.

The original system 3DBT has been declared with the APP "Agence pour la Protection des Programmes" under the Interdeposit Digital number IDDN.FR.001.490023.000.S.P.2006.000.10000. A revised declaration for the latest version of the system is currently being prepared.



Figure 5. The 3D tracker integrates observations from multiple sensors

4.6. Stereo Viewfinder

Participants: Frédéric Devernay [correspondant], Loic Lefort, Elise Mansilla, Sergi Pujades-Rocamora.

Stereoscopy, Auto-calibration, Real-time video processing, Feature matching

This software has been filed with the APP "Agence pour la Protection des Programmes" under the Interdeposit Digital number IDDN.FR.001.370083.000.S.P.2007.000.10000

4.7. Tracking Focus of Attention for Large Screen Interaction

Participants: Rémi Barraquand, Claudine Combe, James Crowley [correspondant], Varun Jain, Sergi Pujades-Rocamora, Lukas Rummelhard.

Embedded Detection and Tracking of Faces for Attention Estimation.

Large multi-touch screens may potentially provide a revolution in the way people can interact with information in public spaces. Technologies now exist to allow inexpensive interactive displays to be installed in shopping areas, subways and urban areas. Such displays can provide location aware access to information including maps and navigation guidance, information about local businesses and commercial activities. While location information is an important component of a users context, information about the age and gender of a user, as well as information about the number of users present can greatly enhance the value of such interaction for both the user and for local commerce and other activities.

The objective of this task is to leverage recent technological advances in real time face detection developed for cell phones and mobile computing to provide a low-cost real time visual sensor for observing users of large multi-touch interactive displays installed in public spaces.

People generally look at things that attract their attention. Thus it is possible to estimate the subject of attention by estimating where people look. The location of visual attention is manifested by a region of space known as the horopter where the optical axis of the two eyes intersect. However estimating the location of attention from human eyes is notoriously difficult, both because the eyes are small relative to the size of the face, and because eyes can rotate in their socket with very high accelerations. Fortunately, when a human attends to something, visual fixation tends to remain at or near that subject of attention, and the eyes are relaxed to a symmetric configuration by turning the face towards the subject of attention. Thus it is possible to estimate human attention by estimating the orientation of the human face.

We have constructed an embedded software system for detecting, tracking and estimating the orientation of human faces. This software has been designed to be embedded on mobile computing devices such as laptop computers, tablets and interactive display panels equipped with a camera that observes the user. Noting the face orientation with respect to the camera makes it possible to estimate the region of the display screen to which the user is attending.

The system uses a Bayesian Particle filter tracker operating on a Scale invariant Gaussian pyramid to provide integrated tracking and estimation of face orientation. The use of Bayesian tracking greatly improves both the reliability and the efficiency for face detection and orientation estimation. The scale invariant Gaussian pyramid provides automatic adaptation to image scale (as occurs with a change in camera optics) and makes it possible to detect and track faces over a large range of distances. Equally important the Gaussian Pyramid provides a very fast computation of a large number of image features that can be used by a variety of image analysis algorithms.

An similar software was released in 2007 using face color rather than appearance. The system SuiviDeCiblesCouleur located individuals in a scene for video communications. FaceStabilisationSystem renormalised the position and scale of images to provide a stabilised video stream. SuiviDeCiblesCouleur has been declared with the APP "Agence pour la Protection des Programmes" under the Interdeposit Digital number IDDN.FR.001.370003.000.S.P.2007.000.21000.

4.8. Visual Emotion Recognition for Health and Well Being.

Participants: Rémi Barraquand, Claudine Combe, James Crowley [correspondant], Varun Jain, Sergi Pujades-Rocamora, Lukas Rummelhard.

Visual Emotion Recognition

People express and feel emotions with their face. Because the face is both externally visible and the seat of emotional expression, facial expression of emotion plays a central role in social interaction between humans. Thus visual recognition of emotions from facial expressions is a core enabling technology for any effort to adapt ICT to improve Health and Wellbeing.

Constructing a technology for automatic visual recognition of emotions requires solutions to a number of hard challenges. Emotions are expressed by coordinated temporal activations of 21 different facial muscles assisted by a number of additional muscles. Activations of these muscles are visible through subtle deformations in the surface structure of the face. Unfortunately, this facial structure can be masked by facial markings, makeup, facial hair, glasses and other obstructions. The exact facial geometry, as well as the coordinated expression of muscles is unique to each individual. In addition, these deformations must be observed and measured under a large variety of illumination conditions as well as a variety of observation angles. Thus the visual recognition of emotions from facial expression remains a challenging open problem in computer vision.

Despite the difficulty of this challenge, important progress has been made in the area of automatic recognition of emotions from face expressions. The systematic cataloging of facial muscle groups as facial action units by Ekman [41] has let a number of research groups to develop libraries of techniques for recognizing the elements of the FACS coding system [33]. Unfortunately, experiments with that system have revealed that the system is very sensitive to both illumination and viewing conditions, as well as the difficulty in interpreting the resulting activation levels as emotions. In particular, this approach requires a high-resolution image with a high signal-to-noise ratio obtained under strong ambient illumination. Such restrictions are not compatible with the mobile imaging system used on tablet computers and mobile phones that are the target of this effort.

As an alternative to detecting activation of facial action units by tracking individual face muscles, we propose to measure physiological parameters that underlie emotions with a global approach. Most human emotions can be expressed as trajectories in a three dimensional space whose features are the physiological parameters of Pleasure-Displeasure, Arousal-Passivity and Dominance-Submission. These three physiological parameters can be measured in a variety of manners including on-body accelerometers, prosody, heart-rate, head movement and global face expression.

The PRIMA Group at Inria has developed robust fast algorithms for detection and recognition of human faces suitable for use in embedded visual systems for mobile devices and telephones. The objective of the work described in this report is to employ these techniques to construct a software system for measuring the physiological parameters commonly associated with emotions that can be embedded in mobile computing devices such as cell phones and tablets.

A revised software package has recently been released to our ICTlab partners for face detection, face tracking, gender and age estimation, and orientation estimation, as part of ICTlabs Smart Spaces action line. This software has been declared with the APP "Agence pour la Protection des Programmes" under the Interdeposit Digital number IDDN.FR.001.370003.000.S.P.2007.000.21000.

A software library, named PrimaCV has been designed, debugged and tested, and released to ICTLabs partners for real time image acquisition, robust invariant multi-scale image description, highly optimized face detection, and face tracking. This software has been substantially modified so as to run on a mobile computing device using the Tegra 3 GPU.

SIROCCO Project-Team

5. Software and Platforms

5.1. Visual Fixation Analysis

Participant: Olivier Le Meur [contact person].

From a set of fixation data and a picture, the software called Visual Fixation Analysis extracts from the input data a number of features (fixation duration, saccade length, orientation of saccade...) and computes an human saliency map. The software can also be used to assess the degree of similarity between a ground truth (eye fixation data) and a predicted saliency map. This software is dedicated to people working in cognitive science and computer vision. This software has been registered at the APP (Agence de Protection des Programmes).

5.2. Hierarchical super-resolution based inpainting

Participant: Olivier Le Meur [contact person].

From an input binary mask and a source picture, the software performs an exemplar-based inpainting. The method is based on the combination of multiple inpainting applied on a low resolution of the input picture. Once the combination has been done, a single-image super-resolution method is applied to recover the details and the high frequency in the inpainted areas. This software is dedicated to people working in image processing and post production. This software is being registered at the APP (Agence de Protection des Programmes).

5.3. Salient object extraction

Participants: Zhi Liu, Olivier Le Meur [contact person].

This software detects salient object in an input picture in an automatic manner. The detection is based on super-pixel segmentation and contrast of histogram. This software is dedicated to people working in image processing and post production. This software is being registered at the APP (Agence de Protection des Programmes).

5.4. loss concealment algorithm using exemplar-based video inpainting

Participants: Ronan Le Boulch, Mounira Ebdelli, Christine Guillemot, Olivier Le Meur [contact person].

This software recovers regions of a video sequence which can be lost after transmission over a network with no guarantee of quality of service. Motion information of impaired areas is first interpolated from the motion vectors of known areas. An exemplar-based video inpainting method is then used to fill in the corrupted areas. This software is being registered at the APP (Agence de Protection des Programmes).

5.5. Standardization

Participants: Christine Guillemot, Laurent Guillo [contact person].

In the continuity of the ADT Picovin-P, we have in 2013, pursued our activities of standardization in the area of multi-view plus depth video coding. We in particular followed the standardization activities within the Joint Collaborative Team on 3D Video Coding Extension (JCT-3V). JCT-3V aims at developing 3D extensions for video codecs, which are AVC (ATM) or HEVC (HTM) based. We have pursued the developments of our proposal related to inter-view motion vector prediction, leading to a joint proposal with Qualcomm and Mediatek which has been adopted in the standard in July 2013.

STARS Project-Team

5. Software and Platforms

5.1. SUP



Figure 5. Tasks of the Scene Understanding Platform (SUP).

SUP is a Scene Understanding Software Platform written in C and C++ (see Figure 5). *SUP* is the continuation of the *VSIP* platform. *SUP* is splitting the workflow of a video processing into several modules, such as acquisition, segmentation, etc., up to activity recognition, to achieve the tasks (detection, classification, etc.) the platform supplies. Each module has a specific interface, and different plugins implementing these interfaces can be used for each step of the video processing. This generic architecture is designed to facilitate:

1. integration of new algorithms in *SUP*;
2. sharing of the algorithms among the Stars team.

Currently, 15 plugins are available, covering the whole processing chain. Several plugins are using the Genius platform, an industrial platform based on VSIP and exploited by Keeneo.

Goals of SUP are twofold:

1. From a video understanding point of view, to allow the Stars researchers sharing the implementation of their work through this platform.
2. From a software engineering point of view, to integrate the results of the dynamic management of vision applications when applied to video analytics.

5.2. ViSEval

ViSEval is a software dedicated to the evaluation and visualization of video processing algorithm outputs. The evaluation of video processing algorithm results is an important step in video analysis research. In video processing, we identify 4 different tasks to evaluate: detection, classification and tracking of physical objects of interest and event recognition.

The proposed evaluation tool (ViSEval, visualization and evaluation) respects three important properties:

- To be able to visualize the algorithm results.
- To be able to visualize the metrics and evaluation results.
- For users to easily modify or add new metrics.

The ViSEval tool is composed of two parts: a GUI to visualize results of the video processing algorithms and metrics results, and an evaluation program to evaluate automatically algorithm outputs on large amount of data. An XML format is defined for the different input files (detected objects from one or several cameras, ground-truth and events). XSD files and associated classes are used to check, read and write automatically the different XML files. The design of the software is based on a system of interfaces-plugins. This architecture allows the user to develop specific treatments according to her/his application (e.g. metrics). There are 6 interfaces:

1. The video interface defines the way to load the images in the interface. For instance the user can develop her/his plugin based on her/his own video format. The tool is delivered with a plugin to load JPEG image, and ASF video.
2. The object filter selects which objects (e.g. objects far from the camera) are processed for the evaluation. The tool is delivered with 3 filters.
3. The distance interface defines how the detected objects match the ground-truth objects based on their bounding box. The tool is delivered with 3 plugins comparing 2D bounding boxes and 3 plugins comparing 3D bounding boxes.
4. The frame metric interface implements metrics (e.g. detection metric, classification metric, ...) which can be computed on each frame of the video. The tool is delivered with 5 frame metrics.
5. The temporal metric interface implements metrics (e.g. tracking metric,...) which are computed on the whole video sequence. The tool is delivered with 3 temporal metrics.
6. The event metric interface implements metrics to evaluate the recognized events. The tool provides 4 metrics.

The GUI is composed of 3 different parts:

1. The widows dedicated to result visualization (see Figure 6):
 - Window 1: the video window displays the current image and information about the detected and ground-truth objects (bounding-boxes, identifier, type,...).
 - Window 2: the 3D virtual scene displays a 3D view of the scene (3D avatars for the detected and ground-truth objects, context, ...).
 - Window 3: the temporal information about the detected and ground truth objects, and about the recognized and ground-truth events.
 - Window 4: the description part gives detailed information about the objects and the events,
 - Window 5: the metric part shows the evaluation results of the frame metrics.
2. The object window enables the user to choose the object to be displayed (see Figure 7).
3. The multi-view window displays the different points of view of the scene (see Figure 8).



Figure 6. GUI of the ViSEvAl software



Figure 7. The object window enables users to choose the object to display



Figure 8. The multi-view window

The evaluation program saves, in a text file, the evaluation results of all the metrics for each frame (whenever it is appropriate), globally for all video sequences or for each object of the ground truth.

The ViSEvAI software was tested and validated into the context of the Cofriend project through its partners (Akka,...). The tool is also used by IMRA, Nice hospital, Institute for Infocomm Research (Singapore),... The software version 1.0 was delivered to APP (French Program Protection Agency) on August 2010. ViSEvAI is under GNU Affero General Public License AGPL (<http://www.gnu.org/licenses/>) since July 2011. The tool is available on the web page : http://www-sop.inria.fr/teams/pulsar/EvaluationTool/ViSEvAI_Description.html

5.3. Clem

The *Clem Toolkit* [68](see Figure 9) is a set of tools devoted to design, simulate, verify and generate code for LE [19] [82] programs. LE is a synchronous language supporting a modular compilation. It also supports automata possibly designed with a dedicated graphical editor.

Each LE program is compiled later into lec and lea files. Then when we want to generate code for different backends, depending on their nature, we can either expand the lec code of programs in order to resolve all abstracted variables and get a single lec file, or we can keep the set of lec files where all the variables of the main program are defined. Then, the *finalization* will simplify the final equations and code is generated for simulation, safety proofs, hardware description or software code. Hardware description (Vhdl) and software code (C) are supplied for LE programs as well as simulation. Moreover, we also generate files to feed the NuSMV model checker [65] in order to perform validation of program behaviors.



Figure 9. The Clem Toolkit

TEXMEX Project-Team

5. Software and Platforms

5.1. Software

When applicable, we provide the IDDN is the official number, which is obtained when registering the software at the APP (Agence de Protection des Programmes).

5.1.1. New Software

5.1.1.1. DeCP-Index

Participants: Laurent Amsaleg [Correspondent], Gylfi Gudmundsson, Diana Moise, Denis Shestakov.

DeCP-Index is a Map-Reduce oriented implementation of the vectorial quantization scheme developed during the PhD of Gylfi Gudmundsson. It is in Java.

First APP deposit: IDDN.FR.001.500011.000.S.P.2013.000.40000

5.1.1.2. DeCP-Scripts

Participants: Laurent Amsaleg [Correspondent], Gylfi Gudmundsson, Diana Moise, Denis Shestakov.

DeCP-Scripts is a series of script for installing, configuring and deploying the Map Reduce framework over the grid infrastructure.

First APP deposit: IDDN.FR.001.500012.000.S.P.2013.000.40000

5.1.1.3. *SVM

Participants: François Poulet [correspondent], Thanh Nghi Doan.

Soon available from the home page of the Inria Parallel Benchmark Suite http://www.irisa.fr/alf/index.php?option=com_content&view=article&id=82&Itemid=&lang=fr.

*SVM include a set of parallel and incremental SVM classifiers for large scale classification tasks on GPU, CPU or cluster / Grid.

5.1.2. Main software started before 2012

5.1.2.1. Peyote

Participants: Sébastien Champion, Jonathan Delhumeau [correspondent], Hervé Jégou.

Peyote is a framework for Video and Image description, indexation and nearest neighbor search. It can be used as-is by a video-search or image-search front-end with the implemented descriptors and search modules. It can also be used via scripting for large-scale experimentation. Finally, thanks to its modularity, it can be used for scientific experimentation on new descriptors or indexation methods. Peyote is used in the AABOT software.

First APP deposit: IDDN.FR.001.420008.000.S.P.2012.000.20900.

5.1.2.2. Aabot

Participant: Jonathan Delhumeau.

AABOT is a tool to facilitate annotation of large video databases. It's primary design focus has been for the annotation on commercials in two 6-month long TV databases. The software keeps a database of already annotated commercials and suggests when it finds a new probable instance. It also validates user annotations by suggesting similar existing commercials if it finds any which are similar by name or content. The user can then confirm the creation of new commercials or accept the correction if he was mistaken.

AABOT is accessed via a web-browser. It is mostly used by uploading and downloading an annotation file. An interactive HTML5 interface is also available when some user feedback is needed (during validation). It uses Peyote as an description / indexing engine.

First APP deposit: IDDN.FR.001.4200010.000.S.P.2012.000.20900.

5.1.2.3. *Pqcodes*

Participant: Hervé Jégou [correspondent].

Jointly maintained with Matthijs Douze, Inria/LEAR.

Pqcodes is a library which implements the approximate k nearest neighbor search method of [88] based on product quantization. This software has been transferred to two companies (in August 2011 and May 2012, respectively).

The current version registered at the APP is IDDN.FR.001.220012.001.S.P.2010.000.10000.

5.1.2.4. *Yael*

Participant: Hervé Jégou [correspondent].

Jointly maintained with Matthijs Douze, from Inria/LEAR.

Yael is a C/python/Matlab library providing (multi-threaded, Blas/Lapack, low level optimization) implementations of computationally demanding functions. In particular, it provides very optimized functions for k-means clustering and exact nearest neighbor search. The library has been downloaded about 2,000 times in 2013.

The current version registered at APP is IDDN.FR.001.220014.001.S.P.2010.000.10000.

5.1.2.5. *BonzaiBoost*

Participant: Christian Raymond [correspondent].

Available at <http://bonzaiboost.gforge.inria.fr/>.

BonzaiBoost stands for boosting over small decisions trees. BonzaiBoost is a general purpose machine-learning program based on decision tree and boosting for building a classifier from text and/or attribute-value data. Currently one configuration of BonzaiBoost is ranked first on <http://mlcomp.org> a website which propose to compare several classification algorithms on many different datasets

5.1.2.6. *Irisa_Ne*

Participant: Christian Raymond [correspondent].

IRISA_NE is a couple of Named Entity tagger, one of them is based on CRF and the other HMM. It is dedicated to automatic transcriptions of speech. It does not take into account uppercase or punctuation and has no concept of sentences. However, they also manage texts with punctuation and capitalization.

5.1.2.7. *IRISA News Topic Segmenter (irints)*

Participants: Guillaume Gravier [correspondent], Pascale Sébillot, Anca-Roxana Simon.

This software is dedicated to unsupervised topic segmentation of texts and transcripts. The software implements several of our research methods and is particularly adapted for automatic transcripts. It provides topic segmentation capabilities virtually for any word-based language, with presets for French, English and German. The software has been licensed to several of our industrial partners.

5.1.3. *Other softwares*

- BAG OF COLORS: describe images based on color
- I-DESCRIPTION: IDDN.FR.001.270047.000.S.P.2003.000.21000
- ASARES: symbolic machine learning system to infer corpus-specific morpho-syntactic and semantic patterns from descriptions of pairs of linguistic elements found in a corpus in which the components are linked by a given semantic relation IDDN.FR.001.0032.000.S.C.2005.000.20900
- ANAMORPHO: detects morphological relations between words in many languages IDDN.FR.001.050022.000.S.P.2008.000.20900
- DIVATEX: audio/video frame server IDDN.FR.001.320006.000.S.P.2006.000.40000

- NAVITEX: video annotation tool IDDN.FR.001.190034.000.S.P.2007.000.40000
- TELEMEX: web service that enables TV and radio stream recording
- VIDSIG: small and robust video signature (64 bits per image)
- VIDSEG: multimodal video segmentation IDDN.FR.001.250009.000.S.P.2009.000.40000
- ISEC: web application used as graphical interface for content-based image search engines
- GPU-KMEANS: k-means algorithm on GPU
- CORRESPONDENCE ANALYSIS: factorial correspondence analysis (FCA) for image retrieval.
- GPU CORRESPONDENCE ANALYSIS: GPU implementation of CORRESPONDENCE ANALYSIS
- CAVIZ: interactive graphical tool to display and extract knowledge from the results of a CORRESPONDENCE ANALYSIS on images
- KIWI: keyword extraction from texts and ASR transcripts
- TOPIC SEGMENTER: topic segmentation of texts and ASR transcripts.
- S2E: automatic discovery of audiovisual structuring events in videos.
- 2PAC: builds classes of words of similar meanings (“semantic classes”) IDDN.FR.001.470028.000.S.P.2006.000.40000

- FAESTOS: Fully Automatic Extraction of Sets of keywords for TOPic characterization and Spotting IDDN.FR.001.470029.000.S.P.2006.000.40000
- FISHNET: automatic Web pages grabber associated with a specific theme
- MATCH MAKER: semantic relation extraction by statistical methods.
- IRISAPHON: grapheme to phoneme conversion
- PYTHON-GEOHASH: implementation of the geometric hashing algorithm [99]
- AVSST: automatic video stream structuring tool (detection of repetitions, classification program/inter-program, EPG alignment) with GUI
- TVSEARCH: content-based retrieval search engine to search and propagate manual annotation such as advertisement in a TV corpora.
- SAMUSA: multimedia content speech/music segmentation
- KERTRACK: visual graphical interface for tracking visual targets based on particle filter tracking or mean-shift.
- MOZAIK2D: spatio-temporal mosaic based on dominant motion compensation.
- BABAZ: audio database management system with an audio-based search function IDDN.FR.001.010006.000.S.P.2012.000.10000
- PIMPY: Python module and binders for multimedia content indexing

5.2. Demonstration: Texmix

Participants: Sébastien Campion [correspondent], Guillaume Gravier.

Structuring a collection of news shows requires some level of semantic understanding of the content in order to segment shows into their successive stories and to create links between stories in the collection, or between stories and related resources on the Web. Spoken material embedded in videos, accessible by means of automatic speech recognition, is a key feature to semantic description of video contents. We have developed multimedia content analysis technology combining automatic speech recognition, natural language processing and information retrieval to automatically create a fully navigable news portal from a collection of video files.

In 2013, we extended the Texmix demonstration to include transcript-free summarization using word discovery.

See the demo at <http://texmix.irisa.fr>.

5.3. Experimental platform

Participants: Laurent Amsaleg, Sébastien Campion [correspondent], Patrick Gros, Pascale Sébillot.

Until 2005, we used various computers to store data and to carry out experiments. In 2005, we began work to specify and set-up dedicated equipment to experiment on very large collections of data. During 2006 and 2007, we specified, bought and installed our first complete platform. It is organized around a very large storage capacity (155TB), and contains 4 acquisition devices (for Digital Terrestrial TV), 3 video servers, and 15 computing servers partially included in the local cluster architecture (IGRIDA). A dedicated website has been developed in 2009 to provide a user support. It contains useful information such as references of available and ready to use software on the cluster, list of corpus stored on the platform, pages for monitoring disk space consumption and cluster loading, tutorials for best practices and cookbooks for treatments of large datasets. In 2010, we have acquired a new large memory server with 144GB of RAM which is used for memory demanding tasks. The previous server dedicated to this kind of jobs (acquired in 2008) has been upgraded to 96GB of RAM. In 2012, we extended our storage capacity to 215TB and expanded our computing resources with two new large memory servers with 256GB of RAM for each of them. Both have their own HPC storage of 12TB. This year our backbone network was fully upgraded in order to connect each element of the platform with a 10GB/s bandwidth.

A new distributed file system architecture was design and will be implement in 2014.

The platform is funded by a joint effort of Inria, INSA Rennes and University of Rennes 1.

5.4. Web services

Participant: Sébastien Campion [correspondent].

This year after a first prototyping of web service where each one of our algorithm was deployed on it's own server, we decided to develop a second version more centralized and named AllGo. AllGo was designed, developed and deployed in order to save resources unnecessarily locked and painful maintenance tasks.

Available at <http://allgo.irisa.fr>, AllGo currently host five TexMex web services (Samusa, Otis, Termex, Nero, VidSeg).

AllGo infrastructure is based on the Ruby On Rails (ROR) framework for the web "frontoffice" part. ROR enable to create and run task with an HTML or XML, JSON API. SideKiq schedule each job on several nodes. Finally, thanks to the new linux container technology named Docker, applications are configured and deployed on agnostic nodes, inside their container. Container must be seen as very light virtual machine. All our application are stored in a private registry. Data are shared with the NFS protocol. A automation software named Puppet manage infrastructure throughout its lifecycle, from provisioning and configuration to orchestration and reporting.

WILLOW Project-Team

5. Software and Platforms

5.1. SParse Modeling Software (SPAMS)

SPAMS v2.4 was released as open-source software in May 2013 (v1.0 was released in September 2009, v2.0 in November 2010). It is an optimization toolbox implementing algorithms to address various machine learning and signal processing problems involving

- Dictionary learning and matrix factorization (NMF, sparse PCA, ...)
- Solving sparse decomposition problems with LARS, coordinate descent, OMP, SOMP, proximal methods
- Solving structured sparse decomposition problems (ℓ_1/ℓ_2 , ℓ_1/ℓ_∞ , sparse group lasso, tree-structured regularization, structured sparsity with overlapping groups,...).

The software and its documentation are available at <http://www.di.ens.fr/willow/SPAMS/>.

5.2. Local dense and sparse space-time features

This is a package with Linux binaries implementing extraction of local space-time features in video. We are preparing a new release of the code implementing highly-efficient video descriptors described in Section 6.4.3. Previous version of the package was released in January 2011. The code supports feature extraction at Harris3D points, on a dense space-time grid as well as at user-supplied space-time locations. The package is publicly available at <http://www.di.ens.fr/~laptev/download/stip-2.0-linux.zip>.

5.3. Automatic Mining of Visual Architectural Elements

The code on automatic mining of visual architectural elements (v4.5) described in (Doersch *et al.* SIGGRAPH 2012) has been publicly released online in January 2013 (earlier version v4.3 was released in December 2012 and v3.0 was released in September 2012) at http://graphics.cs.cmu.edu/projects/whatMakesParis/paris_sigg_release_v4.5.tar.gz.

5.4. Joint learning of actors and actions in video

This is a package of Matlab code implementing the multi-view face processing pipeline and joint learning of actors and actions in movies described in (Bojanowski *et al.* ICCV 2013 [2]). The package was last updated in December 2013 and is available at <http://www.di.ens.fr/willow/research/actoraction/>.

5.5. Visual Place Recognition with Repetitive Structures

Open-source release of the software package for visual localization in urban environments has been made publicly available. The software package implements newly developed method [9] for representing visual data containing repetitive structures (such as building facades or fences), which often occur in urban environments and present significant challenge for current image matching methods. The software is available at http://www.di.ens.fr/willow/research/reptile/download/reptile_demo_ver02.zip.