RESEARCH CENTER

FIELD
Perception, Cognition, Interaction

Activity Report 2012

Section Software

Edition: 2013-04-24

ALPAGE Project-Team

5. Software

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. System 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 [120].

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 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. [117], [118], and very large amount of data (700 millions of words) in the SCRIBO project.

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 [122]. 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 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. A new version is under development to provide an even more compact syntax and some checking mechanisms to avoid frequent typo errors.

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 Candito [correspondant], Djamé Seddah.

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

Alpage has developed as support of the research papers [74], [65], [66], [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. The parser is available under a GPL license.

5.5. The MICA parser

Participants: Benoît Sagot [correspondant], Marie Candito, Pierre Boullier, Djamé Seddah.

Web page:
http://mica.lif.univ-mrs.fr/

MICA (Marseille-Inria-Columbia- AT&T) is a freely available dependency parser [57] 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 [97] 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...).

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 (ii) 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;
- Development of new local grammars for detecting new types of entities, such as chemical formulae or dimensions, in the context of the PACTE project.

5.7. MElt

Participants: Benoît Sagot [correspondant], Pascal Denis.

MElt is a part-of-speech tagger, initially trained for French (on the French TreeBank and coupled with the Le*fff*), English [78], Spanish, Kurmanji Kurdish [125] and Persian [106], [107]. It is state-of-the-art for French. It is distributed freely as a part of the Alpage linguistic workbench.

In 2012, MElt has underwent two major upgrades:

- It has been successfully trained and used on Italian [35], Spanish [26] and German data. In particular, a statistical parsing architecture for Italian that used MElt in a pre-processing step has obtained the best results in the EVALITA shared task on Italian parsing [35].
- MElt can now be called within a wrapper developed for handling noisy textual data such as user-generated content produced on Web 2.0 platforms (forums, blogs, social media); more precisely, this wrapper is able to "clean" such data, then tag it using MElt, and finally transfer MElt annotations from the "cleaned" data, which could be annotated more easily, to the original noisy data. This architecture has proved useful on French for creating the French Social Media Bank [37], [36]. On English, it has played an important role within both variants of the Alpage parsing architecture that were ranked 2nd and 3rd at the SANCL shared task on parsing user-generated content, organized by Google [38].

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 Le*fff*, a morphological and syntactic lexicon for French.

Historically, the Le*fff* 1 was a freely available French morphological lexicon for verbs that has been automatically extracted from a very large corpus. Since version 2, the Le*fff* 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 Le*fff*. The version 3 of the Le*fff*, 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 Le*ff*e), Polish, Slovak, English, Galician, Persian, Kurdish, Italian and since this year for German, as well as for Latin and Maltese verbs. These lexicons are used in various tools, including instances of the MElt POS-tagger.

Alexina also hosts *Aleda* [114], [33] 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], Marion Richard, 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 [99], [100]. 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 [101].

2012 results concerning the WOLF are described in the corresponding section.

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

5.10. Automatic construction of distributional thesauri

Participant: Enrique Henestroza Anguiano [correspondant].

FREDISTis 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, FREDISTis 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.

5.11. 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).

5.12. System EasyRef

Participants: Éric Villemonte de La Clergerie [maintainer], Corentin Ribeyre.

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 PASSAGE.

A preliminary version 3 has been developed by François Guérin and revised by Éric de La Clergerie, relying on Berkeley DB XML to handle very large annotated corpora and to provide a complete query language expanded as XQuery expressions. EASYREF is maintained under Inria GForge.

METISS Project-Team

5. Software

5.1. Audio signal processing, segmentation and classification toolkits

Participant: Guillaume Gravier.

Guillaume Gravier is now with the TEXMEX group but this software is being used by several members of the METISS group.

speech, audio, signal, analysis, processing, audio stream, detection, tracking, segmentation, audio indexing, speaker verification

The SPro toolkit provides standard front-end analysis algorithms for speech signal processing. It is systematically used in the METISS group for activities in speech and speaker recognition as well as in audio indexing. The toolkit is developed for Unix environments and is distributed as a free software with a GPL license. It is used by several other French laboratories working in the field of speech processing.

In the framework of our activities on audio indexing and speaker recognition, AudioSeg, a toolkit for the segmentation of audio streams has been developed and is distributed for Unix platforms under the GPL agreement. This toolkit provides generic tools for the segmentation and indexing of audio streams, such as audio activity detection, abrupt change detection, segment clustering, Gaussian mixture modeling and joint segmentation and detection using hidden Markov models. The toolkit relies on the SPro software for feature extraction.

Contact : guillaume.gravier@irisa.fr
http://gforge.inria.fr/projects/spro, http://gforge.inria.fr/projects/audioseg

5.2. Irene: a speech recognition and transcription platform

Participant: Guillaume Gravier.

Guillaume Gravier is now with the TEXMEX group but this software is being used by several members of the METISS group.

speech modeling, speech recognition, broadcast news indexing, beam-search, Viterbi, HMM

In collaboration with the computer science dept. at ENST, METISS has actively participated in the past years in the development of the freely available Sirocco large vocabulary speech recognition software [91]. The Sirocco project started as an Inria Concerted Research Action now works on the basis of voluntary contributions.

The Sirocco speech recognition software was then used as the heart of the transcription modules whithin a spoken document analysis platform called IRENE. In particular, it has been extensively used for research on ASR and NLP as well as for work on phonetic landmarks in statistical speech recognition.

In 2009, the integration of IRENE in the multimedia indexing platform of IRISA was completed, incorporating improvements benchmarked during the ESTER 2 evaluation campaign in december 2008. Additionnal improvements were alos carried out such as bandwidth segmentation and improved segment clustering for unsupervised acoustic model adaptation. The integration of IRENE in the multimedia indexing platform was mainly validated on large datasets extracted from TV streams.

Contact : guillaume.gravier@irisa.fr
http://gforge.inria.fr/projects/sirocco

5.3. MPTK: the Matching Pursuit Toolkit

Participants: Rémi Gribonval, Jules Espiau.

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.

MPTK has been entirely developed within the METISS group mainly to overcome limitations of existing Matching Pursuit implementations in terms of ease of maintainability, memory footage or computation speed. One of the aims is to be able to process in reasonable time large audio files to explore the new possibilities which Matching Pursuit can offer in speech signal processing. With the new implementation, it is now possible indeed to process a one hour audio signal in as little as twenty minutes.

Thanks to an Inria software development operation (Opération de Développement Logiciel, ODL) started in September 2006, METISS efforts have been targeted at easing the distribution of MPTK by improving its portability to different platforms and simplifying its developpers' API. Besides pure software engineering improvements, this implied setting up a new website with an FAQ, developing new interfaces between MPTK and Matlab and Python, writing a portable Graphical User Interface to complement command line utilities, strengthening the robustness of the input/output using XML where possible, and most importantly setting up a whole new plugin API to decouple the core of the library from possible third party contributions.

Collaboration : Laboratoire d'Acoustique Musicale (University of Paris VII, Jussieu).

Contact : remi.gribonval@irisa.fr

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

5.4. FASST

Participants: Emmanuel Vincent [correspondant], Nancy Bertin, Frédéric Bimbot.

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

5.5. NACHOS

Participants: Nancy Bertin [correspondant], 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 [38], 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.

PAROLE Project-Team

5. Software

5.1. WinSnoori

contact : Yves Laprie (Yves.Laprie@loria.fr)

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 Snoori 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 main improvement concerns automatic formant tracking which is now available with other tools for copy synthesis. It is now possible to determine parameters for the formant synthesizer of Klatt quite automatically. The first step is formant tracking, then the determination of F0 parameters and finally the adjustment of formant amplitudes for the parallel branch of the Klatt synthesizer enable a synthetic speech signal to be generated. The automatic formant tracking that has been implemented is an improved version of the concurrent curve formant tracking [49]. One key point of this tracking algorithm is the construction of initial rough estimates of formant trajectories. The previous algorithm used a mobile average applied onto LPC roots. The window is sufficiently large (200 ms) to remove fast varying variations due to the detection of spurious roots. The counterpart of this long duration is that the mobile average prevents formants fairly far from the mobile average to be kept. This is particularly sensitive in the case of F2 which presents low frequency values for back vowels. A simple algorithm to detect back vowels from the overall spectral shape and particularly energy levels has been added in order to keep extreme values of F2 which are relevant.

Together with other improvements reported during the last years, formant tracking enables copy synthesis. The current version of WinSnoori is available on http://www.winsnoori.fr.

5.2. JSnoori

contact : Yves Laprie (Yves.Laprie@loria.fr)

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. Xarticulator

contact : Yves Laprie (Yves.Laprie@loria.fr)

Xarticulators software is intended to delineate contours of speech articulators in X-ray images, construct articulatory models and 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 designed an algorithm 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. In addition we also added the possibility of processing digitized manual delineation results made on sheet of papers when no software was available

5.4. SUBWEB

contacts : David Langlois (langlois@loria.fr) and Kamel Smaïli (smaili@loria.fr).

We published in 2007 a method which allows to align sub-titles comparable corpora [50]. 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 and a graphical representation of the course of the algorithm. This work has been supported by CPER/TALC/SUBWEB [2].

5.5. SELORIA

contact : Odile Mella (Odile.Mella@loria.fr).

SELORIA is a toolbox for speaker diarization.

The system contains the following steps:

- Speaker change detection: to find points in the audio stream which are candidates for speaker change points, a distance is computed between two Gaussian modeling data of two adjacent given-length windows. By sliding both windows on the whole audio stream, a distance curve is obtained. A peak in this curve is thus considered as a speaker change point.
- Segment recombination: too many speaker turn points detected during the previous step results in a lot of false alarms. A segment recombination using BIC is needed to recombine adjacent segments uttered by the same speaker.
- Speaker clustering: in this step, speech segments of the same speaker are clustered. Top-down clustering techniques or bottom-up hierarchical clustering techniques using BIC can be used.
- Viterbi re-segmentation: the previous clustering step provides enough data for every speaker to estimate multi-gaussian speaker models. These models are used by a Viterbi algorithm to refine the boundaries between speakers.
- Second speaker clustering step (called cluster recombination): This step uses Universal Background Models (UBM) and the Normalized Cross Likelihood Ratio (NCLR) measure.

This toolbox is derived from mClust designed by LIUM.

5.6. ANTS

contacts : Dominique Fohr (fohr@loria.fr) and Denis Jouvet (denis.jouvet@inria.fr).

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.

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

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.

5.7. JTrans

Contact : Christophe Cerisara (Christophe.Cerisara@loria.fr).

JTrans is an open-source software for semi-automatic alignement of speech and textual corpus. It is written 100% in JAVA and exploits libraries developed since several years in our team. Two algorithms are available for automatic alignment: a block-viterbi and standard forced-alignement Viterbi. The latter is used when manual anchors are defined, while the former is used for long audio files that do not fit in memory. It is designed to be intuitive and easy to use, with a focus on GUI design. The rationale behind JTrans is to let the user control and check on-the-fly the automatic alignment algorithms. It is bundled for now with a French phonetic lexicon and French models.

Recent improvements include its integration within the JSafran platform and its release as a Java applet that can be demonstrated on web pages. During the last three months, JTrans has been downloaded about 120 times and seven users of JTrans, outside LORIA, have directly contacted the team for requests about JTrans.

JTrans is developed in the context of the CPER MISN TALC project, in collaboration between the Parole and Talaris Inria teams, and CNRS researchers from the ATILF laboratory. It is distributed under the Cecill-C licence, and can be downloaded at http://synalp.loria.fr/?n=Research.Software

5.8. CoALT

contacts : Dominique Fohr (dominique.fohr@loria.fr) and Odile Mella (odile.mella@loria.fr).

CoALT (Comparing Automatic Labelling Tools) compares two automatic labellers 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 labelling result and the hand-labelling one.

5.9. TTS SoJA

contact : Vincent Colotte (Vincent.Colotte@loria.fr).

TTS SoJA (Speech synthesis platform in Java) is a software of text-to-speech synthesis system. 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 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

5.10. Corpus Recorder

contact : Vincent Colotte (Vincent.Colotte@loria.fr).

Corpus Recorder is a software for the recording of audio corpora. It provides a easy tool to record with a microphone. The gain of the audio input 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 to display a textual context of the sentence to pronounce. This software is suitable for recording sentences with information to guide the speaker.

The software is developed in Tcl/Tk (tested under Windows and Linux). It was used for the recording of sentences for the TTS system SOJA and during the Intonale Project (Prosody Modeling).

5.11. VisArtico

contact : Slim Ouni (Slim.Ouni@loria.fr).

VisArtico is intended to visualize articulatory data acquired using an articulograph [30], [29]. 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 cross-platform software as it is developed in JAVA and does not need any external library or framework to be additionally installed. 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.

SEMAGRAMME Team

5. Software

5.1. Leopar

Participants: Bruno Guillaume [correspondant], Guy Perrier.

Interaction Grammar, parsing

5.1.1. Software description

Leopar is a parser for natural languages which is based on the formalism of Interaction Grammars [30]. 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, Leopar uses some specific and non-trivial data-structures and algorithms.

The electrostatic principle has been intensively considered in Leopar. 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 $f \longleftarrow v$ by a dual positive feature-value pair $f \longrightarrow v$. 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 developped 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

Leopar is presented and documented at http://leopar.loria.fr; an online demonstration page can be found at http://leopar.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,

5.2. ACG Development Toolkit

Participants: Sylvain Pogodalla [correspondant], Philippe de Groote.

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 the ACG development toolkit prototype [1] 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.

Enabling transformation from the object terms to the abstract terms is the first step of future development for the ACG support system. A parsing algorithm based on [32]'s methods 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 requires further adaptations. 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 [correspondant], 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 or 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.
- 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).

[1] Available at http://acg.gforge.inria.fr with a CeCILL license.

A demonstration of the Grew Software was presented at the TALN conference in june in Grenoble.[15]

The Grew software were 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 [12], [14]. More recently, it was used on the Sequoia TreeBank, to produce deep syntax annotation and DMRS Semantic representations.

Another application of the Grew software which is currently investigated is the detection of annotation errors in corpora. Graph Rewriting is use to detect ill-formed structures that don't fit the annotation guide requirements. In collaboration with the Alpage team, this was applied to the Sequoia Corpus and the reported errors were corrected in version 3.2 and 3.3 of the corpus[2].

5.4. Other developments

Participant: Bruno Guillaume [correspondant].

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

- A concordancer named CONDOR which is usable online: http://condor.loria.fr. With Condor, ut 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 presented in http://dep2pict.loria.fr; it is usable online http://dep2pict.loria.fr/demo.

[2] https://www.rocq.inria.fr/alpage-wiki/tiki-index.php?page=CorpusSequoia

ALICE Project-Team

4. Software

4.1. Graphite

Participants: Dobrina Boltcheva, Phuong Ho, Bruno Lévy, David Lopez, Romain Merland, Vincent Nivoliers, Jeanne Pellerin, Nicolas Ray.

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 (1000 downloads in two months). Graphite is one of the common software platforms used in the frame of the European Network of Excellence AIMShape .

4.2. MicroMegas

Participant: Samuel Hornus.

Micromegas 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 "GraphiteLife-Explorer". Biologists need simple spatial modeling tools to help in understanding the role of objects’ relative position 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.

In 2012, its development has been actively pursued by Samuel Hornus in the first trimester. The software is freely downloadable. A paper describing will appear in the broad journal PLOS One [9]. A poster was also presented at the European Conference on Computational Biology in september 2012.

4.3. CGAL package for Delaunay triangulations

Participant: Samuel Hornus.

This year was devolved also to finishing touches on the CGAL package for Delaunay triangulations (3rd submission to the CGAL editorial board).

Following the reviews for the second submission, Samuel Hornus has collaborated with Olivier Devillers (Inria Sophia Antipolis) to put the finishing touches to a new CGAL package for Delaunay triangulation in any dimension. It provides exact construction of Delaunay triangulations, supporting both the addition and deletion of vertices. The code takes the form of a collection of C++ template classes to ensures high performance when specializing the code to a given euclidian dimension.

4.4. OpenNL - Open Numerical Library

Participants: Thomas Jost, 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 CGALpackage for mesh parameterization.
- Concurrent Number Cruncher: this software library extends OpenNL with parallel computing on the GPU, implemented using the CUDA API.

4.5. Intersurf

Participants: Xavier Cavin, Nicolas Ray.

Intersurf is a plugin of the VMD (Visual Molecular Dynamics) software. VMD is developed by the Theoretical and Computational Biophysics Group at the Beckmann Institute at University of Illinois. The Intersurf plugin is released with the official version of VMD since the 1.8.3 release. It provides surfaces representing the interaction between two groups of atoms, and colors can be added to represent interaction forces between these groups of atoms. We plan to include in this package the new results obtained this year in molecular surface visualization by Matthieu Chavent.

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 and the Alice / Inria Nancy Grand-Est teams.

AVIZ Project-Team

5. Software

5.1. Graph Cuisine

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

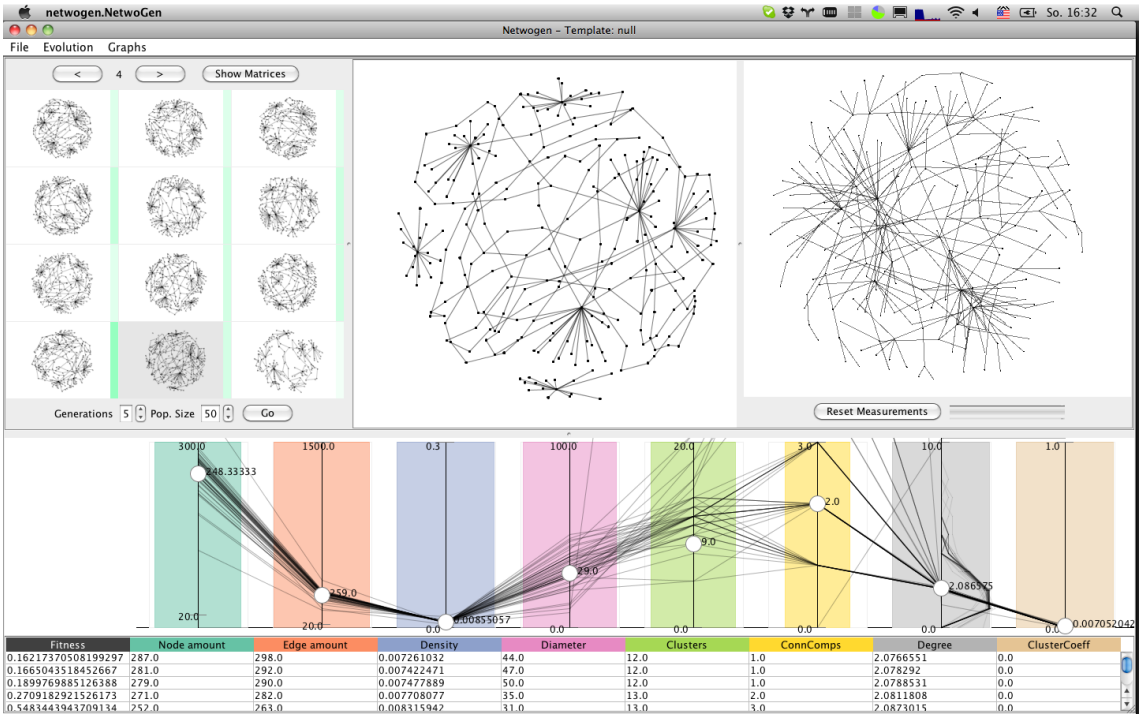


Figure 1. GraphCuisine interface showing one real graph (right), the measures extracted from it (bottom circles), several graphs with similar measures (left) and one of them selected (middle).

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.

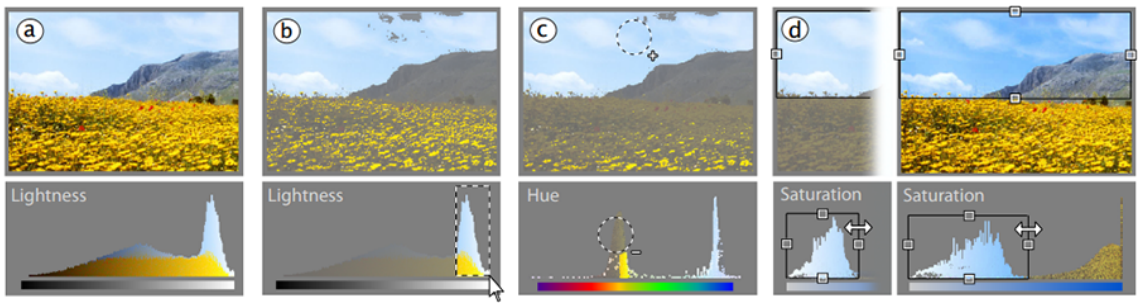


Figure 2. Example of sky enhancement with Histomages: (a) the image is duplicated and its pixels rearranged into a lightness histogram ; (b) bright pixels are selected with the rubber-band selection tool; (c) all pixels are rearranged into a hue histogram and yellow pixels are filtered out with the subtract 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.

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 [29]. http://www.aviz.fr/histomages/.

5.3. Gliimpse

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

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

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 Hémery, Jean-Daniel Fekete [correspondant].

Information Visualization, Java, Toolkit

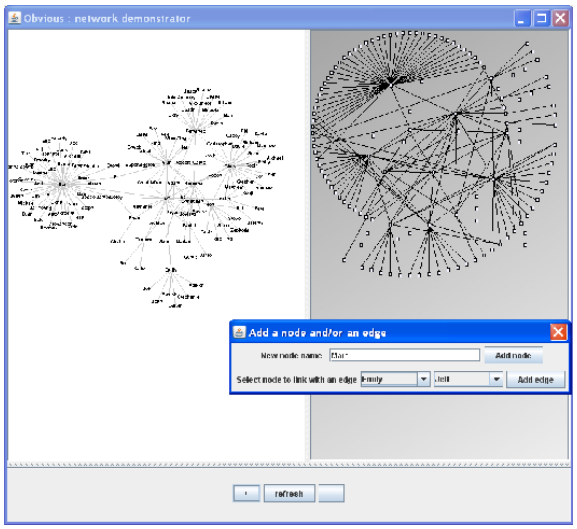


Figure 4. The Obvious toolkit showing the same graph with a Prefuse and an IVTK rendering.

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 final 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) [53]. 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.

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

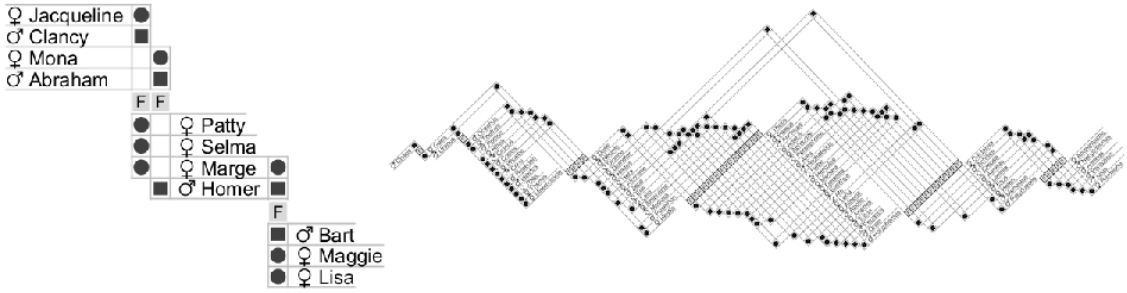


Figure 5. The genealogy of the Simpsons family (left) and of the Greek Pantheon (right), produced by the GeneaQuilts software.

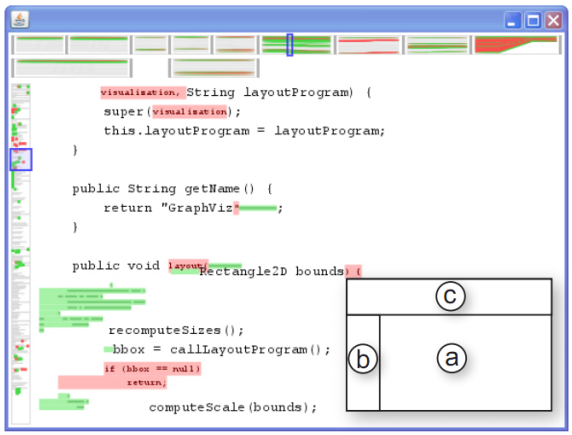


Figure 6. Screenshot the Diffamation system during a transition: (a) the document view, (b) the overview scrollbar and (c) the timeline.

The 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

The InfoVis Toolkit [5] 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.

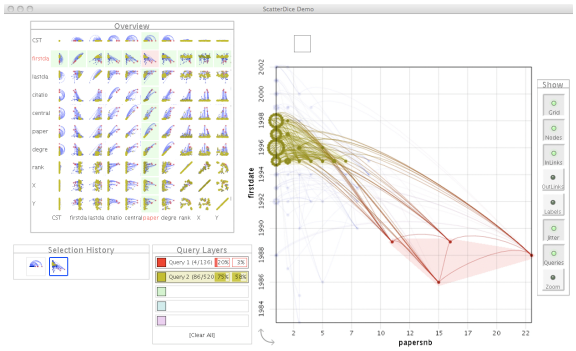


Figure 7. Screenshot the GraphDice system.

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) [4]: 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.

IMAGINE Team

5. Software

5.1. MyCorporisFabrica

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

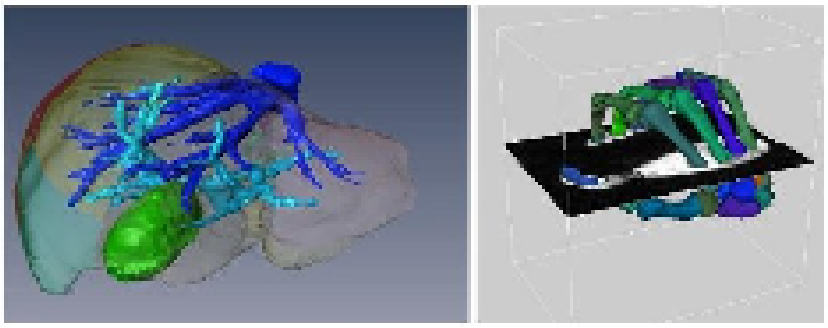


Figure 1. My Corporis Fabrica is an anatomical knowledge database developed in our team.

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.

5.2. SOFA

Participants: Guillaume Bousquet, Ali Hamadi Dicko, François Faure, François Jourdes.

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.

5.3. Convol

Participants: Marie-Paule Cani, Amaury Jung, Galel Koraa, Maxime Quiblier, Cédric Zanni.

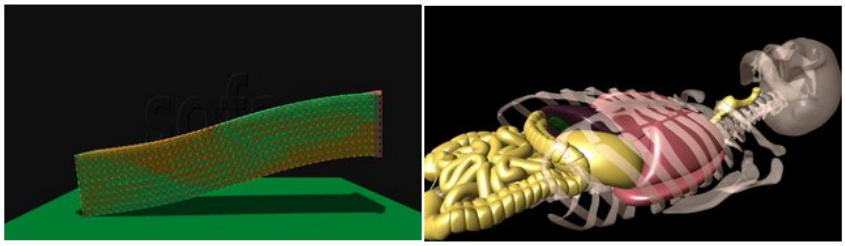


Figure 2. SOFA is an open source simulator for physically based modeling.

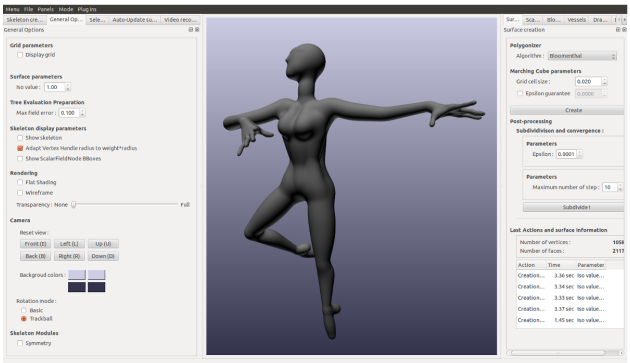


Figure 3. Example of implicit surface and the GUI proposed in the Convol software.

Convol is a new C++ library we develop for easing our work on implicit surfaces – and more particularly on the sub-class of convolution surfaces. It enables us to make our latest research results soon available to the rest of the group and easily usable in our industrial partnerships. Convol incorporates all the necessary material for constructive implicit modeling: skeleton-based distance and convolution 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 skeleton or through sketch based modeling.
This development is funded by Inria as support to our research group.

IN-SITU Project-Team

5. Software

5.1. jBricks

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

jBricks (Figure 1) is a Java toolkit that integrates a high-quality 2D graphics rendering engine based on ZVTM (section 5.2) and a versatile input configuration module (based on ICon [45] 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.

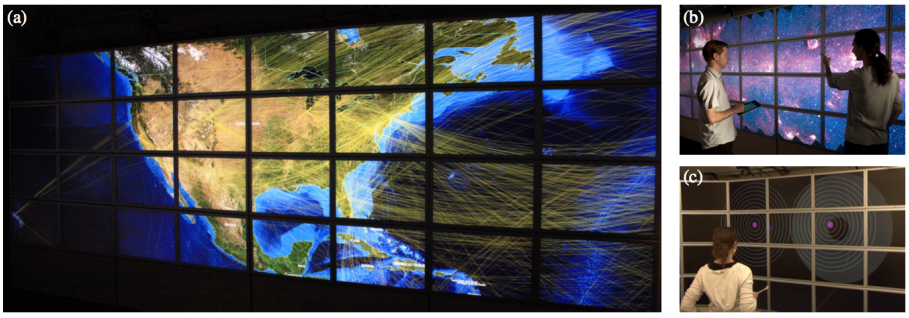


Figure 1. jBricks applications running on the WILD platform (32 tiles for a total resolution of 20 480 × 6 400 pixels). (a) Zoomed-in visualization of the North-American part of the world-wide air traffic network (1 200 airports, 5 700 connections) overlaid on NASA's Blue Marble Next Generation images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

- 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

5.2. The Zoomable Visual Transformation Machine

Participants: Caroline Appert, Rodrigo de Almeida, Olivier Chapuis, Arjit Gupta, Emmanuel Pietriga [correspondant], Mathieu Nancel, Romain Primet.

ZVTM provides application programmers with building blocks for implementing complex multi-scale interface components that cannot be handled by traditional WIMP widgets. Featuring off-the-shelf visualisation and navigation components that are easy to combine, ZVTM provides a simple yet powerful API and handles low-level operations such as multi-threading, clipping, repaint requests and animation management. The toolkit is based on the metaphor of universes that can be observed through smart movable/zoomable cameras. The graphical object model permits management of a large number of complex geometrical shapes. It emphasizes perceptual continuity via an advanced animation module that can animate virtually any on-screen modification. This ranges from camera movements and activation of distortion lenses to modification of the visual variables of graphical objects. Various temporal pacing functions are available to control the execution of these animations. ZVTM is now one of the core components of our jBricks toolkit for wall-sized displays (Section 5.1), and current development activities around the toolkit focus on making applications run transparently on cluster-driven ultra-high-resolution wall-sized displays such as that of the WILD visualization platform. The toolkit is also used to develop advanced visualization components for the ALMA observatory's operations monitoring and control software [26].

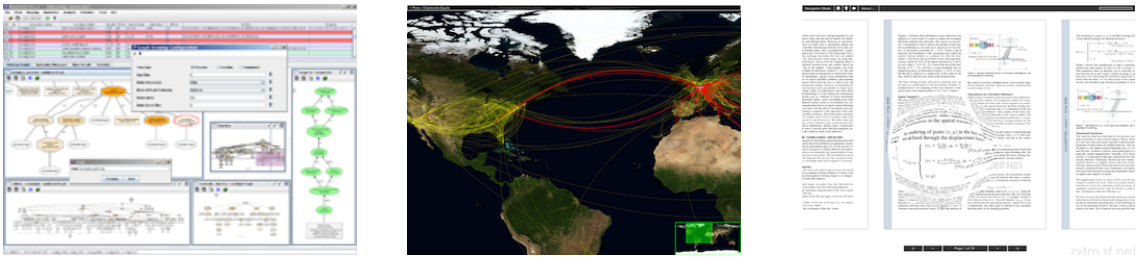


Figure 2. ZVTM used in various applications

Initially developed by Xerox Research Centre Europe and the World Wide Web Consortium (W3C) team at MIT, ZVTM has been available as open-source software under the GNU Lesser General Public License (LGPL) since early 2002. It is used in both academic and industrial projects such as IsaViz (http://www.w3.org/2001/11/IsaViz/), W3C's visual browser/editor for RDF, Blast2GO (Figure 2 - left) (http://www.blast2go.org/), or ZGRViewer (http://zvtm.sourceforge.net/zgrviewer.html) for viewing large graphs generated by AT&T GraphViz [1] (Figure 2 - right). The development of the toolkit is now supported by Inria. More information can be found at http://zvtm.sourceforge.net and [52] and [51].

- ACM: H.5.2 [User Interfaces]: Graphical user interfaces (GUI)
- Software benefit: See Pietriga, A Toolkit for Addressing HCI Issues in Visual Language Environments, IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC '05), pages 145-152, September 2005
- License: LGPL
- Type of human computer interaction: Graphique
- OS/Middleware: Java (Linux, Mac OS X, Windows)
- Required library or software: several, managed through Maven
- Programming language: Java

5.3. The SwingStates Toolkit

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

[1] http://www.graphviz.org

SwingStates [37] 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 [38], 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 [37] and ICon [45].

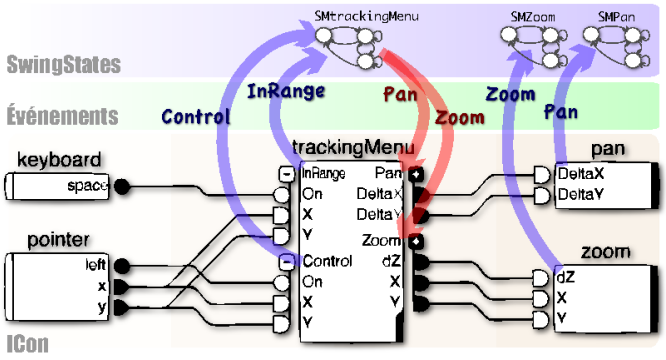


Figure 4. State machines and data flow in FlowStates

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 [correspondant], Michel Beaudouin-Lafon, Wendy Mackay.

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.

With the *Touchstone design platform*, a user specifies the factors and the measures of the experiment, the blocking and counterbalancing of trials, and assess the time it will take to run the experiment. Multiple designs can be explored in parallel to assess the various trade-offs. The output of the design platform is an XML file that can be used as input for the run platform.

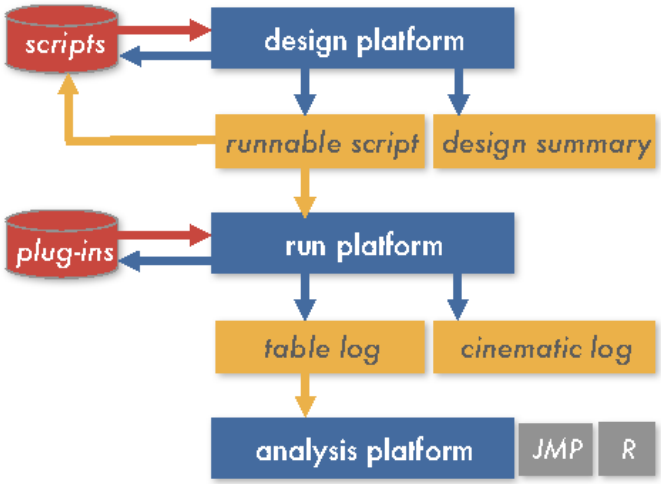


Figure 5. The architecture of the Touchstone platform

The *Touchstone run platform* provides a framework to implement and run an experiment and to collect experimental data. It uses a flexible plug-in architecture to manage a variety of input devices and interaction techniques. The runs of the experiment are controlled by an XML script that can be produced by the design platform.

The analysis platform currently consists of data analysis tools such as JMP, R or Excel. Log data produced by the run platform can be directly loaded into any of these tools. In a future version, analysis sketches will be derived from the experimental design to assist with the analysis.

Touchstone has been used heavily at INSITU over the past three years for the many experiments that we design and run. It has also been used for teaching for the first time in 2011. Students used it to design various experiments during tutorial classes in Master 2 Interaction ("Introduction to HCI" module).

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 [43] 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 [54]. This year we used again Metisse to implement novel desktop interaction techniques [4].

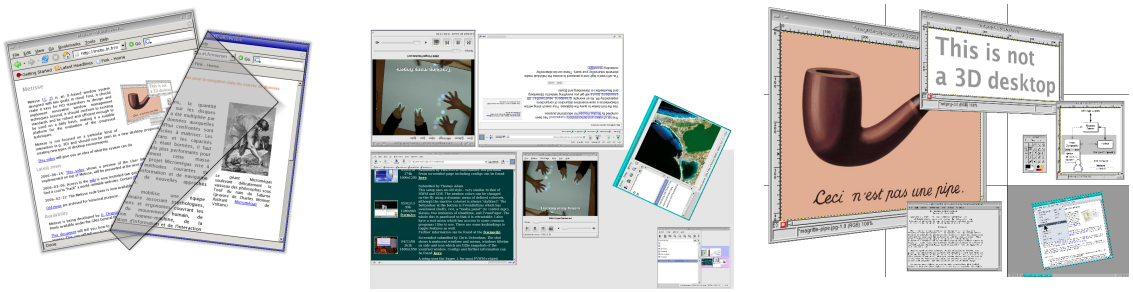


Figure 6. Sample window management techniques implemented with Metisse: extended paper metaphor (left), interactive table configuration that allows to duplicate and rotate windows (middle) and zoomable 3D desktop (right).

- Web: http://insitu.lri.fr/metisse/
- ACM: H.5.2 [User Interfaces]: Windowing systems
- Software benefit: see [43], [54], [44], [47] and [4].
- License: GPL
- Type of human computer interaction: Graphique
- OS/Middleware: X Window et Mac OS X
- Required library or software: OpenGL via nucleo [2] and some usual C/C++ libraries
- Programming language: * C/C++

5.7. Wmtrace

Participant: Olivier Chapuis [correspondant].

Wmtrace [42] includes two tools that help us study an individual user's window management activity. The first tool runs in the background of an X Window session and continuously logs information about windows and how they are being manipulated. The second uses a VCR-like interface (Figure 7) to replay the resulting logs and analyze the entire session. This tool provides several ways to filter the logs and extract high-level information, including interactive move events and mouse speed. Both tools allow HCI researchers to perform qualitative and quantitative statistical analyses of window management activity.

- Web: http://insitu.lri.fr/~chapuis/software/wmtrace/.
- ACM: H.5.2 [User Interfaces]: Windowing systems
- Software benefit: see [42], [47], [41].
- License: GPL

[2] http://interaction.lille.inria.fr/~roussel/projects/nucleo/index.html

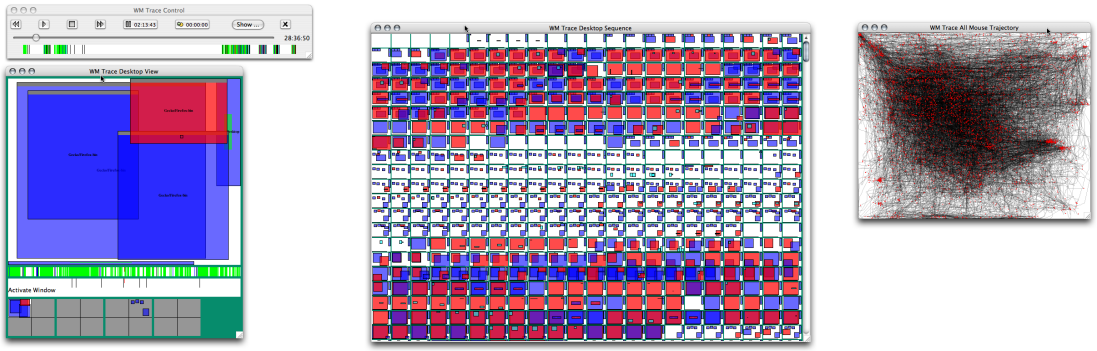


Figure 7. VCR-like interface, session overview and sample plots of mouse trajectories (black) and mouse clicks (red)

- Type of human computer interaction: Deamon and Graphique
- OS/Middleware: X Window (deamon) and Java (VCR interface)
- Required library or software: all X libraries (daemon) and Java (VCR interface)
- Programming language: * C and Java

5.8. The Substance Middleware

Participants: Michel Beaudouin-Lafon [correspondant], Clemens Klokmose, 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 [48]. 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 [53].

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 8): 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)

Figure 8. The Canvas (left) and SubstanceGrise (right) applications developed with Substance. (©CNRS-Phototheque - Cyril FRESILLON for SubstanceGrise).

- Software benefit: See T. Gjerlufsen, C. Klokmose, 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

5.9. 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 [46]. 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 [39]. Scotty currently runs on Mac OS X for applications written with the Cocoa user interface framework.

Scotty has been used to create a number of extensions (Figure 9). *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.

Figure 9. 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).

- OS/Middleware: Mac OS X
- Required library or software: none
- Programming language: Objective-C, Python

MANAO Team

4. Software

4.1. EIGEN

Participants: G. Guennebaud, D. Nuentsa

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 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 and 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 20000 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 3.1 version in June 2012. Eigen is currently supported by Inria through an ADT started in January 2012.

Facts:

- Web: http://eigen.tuxfamily.org/
- License: LGPL3+

MAVERICK Team

5. Software

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. 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.

5.3. 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 developpement 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.

5.4. MobiNet

Participants: Fabrice Neyret [contact], Joëlle Thollot.

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 scool class in the frame of the courses. Moreover, presentation in workshops and institutes are done, and a web site repository is maintained.

5.5. Freestyle

Freestyle is a software for Non-Photorealistic Line Drawing rendering from 3D scenes (Figure 2). 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 lead to two publications [23], [24].

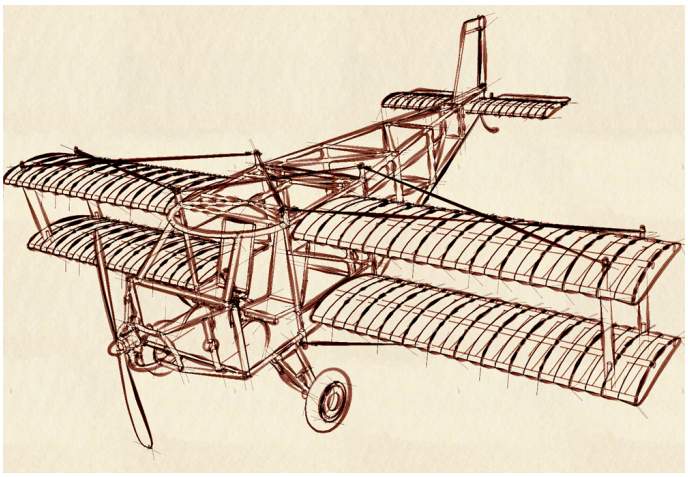


Figure 2. Stylized plane using Freestyle.

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.

5.6. Diffusion Curves

Participant: Joëlle Thollot [contact].

We provide an implementation of the vector drawing tool described in the 2008 Diffusion Curves Siggraph paper (Figure 3) 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.

5.7. VRender: vector figures

Participant: Cyril Soler [contact].

[1] http://www.blender.org/

Figure 3. Diffusion curves freely downloadable demo.

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.

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.

5.8. 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 transfered to companies. Eric Bruneton was hired by Google-Zürich in september 2011, but will be able to keep some participation in the project.

5.9. GigaVoxel

Participants: Fabrice Neyret [contact], Morgan Armand, Eric Bruneton, 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 4). The goal of this platform is the real-time rendering of very large very detailed scenes. Performances permit showing details over deep zooms and walk through very crowdy scenes (which are rigid, for the moment). The principle is GPU ray-tracing of volumetric-encoded multiscale data with minimal just-in time generation of data (accounting visibility and needed resolution) kept in a cache on GPU. The representation eases the cheap management of soft shadows, depth of field, anti-aliasing and geometric LOD. Beside the representation, data management and base rendering algorithm themself, we also worked on realtime light transport, and on quality prefiltering of complex data. This work led to numerous publications ([22], [21], [20]). Several licences have been sold to companies. we also did a technical presentation of the GigaVoxels tool during Afig conference [17] in order to invit the community to use the tool.

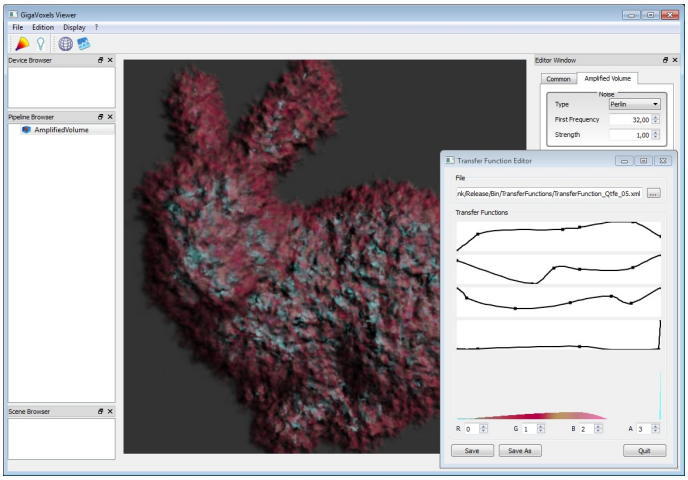


Figure 4. GigaVoxels freely downloadable demo.

MIMETIC Team

5. Software

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

Participants: Stéphane Donikian [contact], Fabrice Lamarche [contact].

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 (sequence, parallelism, loops, alternatives...). Theses 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.

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 obtenained from our off-line tool (that tranforms 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].

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

5.1. LibGINA

Participant: Laurent Grisoni [correspondant].

This library has been developped 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: Paolo Olivo, Nicolas Roussel [correspondant], Ibrahim Yapici.

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.

Current version: 0.1 - June 2011 (IDDN.FR.001.270005.000.S.P.2011.000.10000)

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 scheduled to be publicly released in 2012, the licence remaining to be decided.

Web site: http://libpointing.org/

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

POTIOC Team

5. Software

5.1. OpenViBE

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

As part of our research work on BCI, we contribute to the development of the OpenViBE [4] software, which is an open source platform dedicated to the design, evaluation and use of BCI for real and virtual applications. OpenViBE development is leaded by Inria, and Potioc will be one of the Inria team contributing to its evolution. Moreover, Potioc is implied in an Inria ADT (Technological Development Action) project that has just started, an that is dedicated to the development of OpenViBE together with 3 other Inria teams (Hybrid, Athena, Cortex).

5.2. Drile

Participant: Florent Berthaut.

As part of the research on Virtual Reality for Musical Performance, notably the Drile system, various software are being developped 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

[4] http://openvibe.inria.fr

REVES Project-Team

4. Software

4.1. RID: Rich Intrinsic Decomposer

Participants: Pierre-Yves Laffont, Adrien Bousseau, George Drettakis.

We developed a software platform to perform rich intrinsic decomposition methods from photographs of outdoor scenes, as described in [18]. 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.

4.2. Imerse: Inria Multi-Environment Realistic Simulation Engine

Participants: Adrien David, George Drettakis.

In the context of the ADT Interact3D and the ARC NIEVE, we developed Imerse, a middleware to be used as a VR engine, helping in the implementation of realistic simulations for immersive installations. Imerse provides a wrapper to OSG's (OpenSceneGraph) deep scene graph and its traversals abilities into an abstracted collection of high level objects which directly represent realistic entities (such as indoor elements, machines and realistic characters). It provides capacities such as skeletal animations or spatialized audio by interfacing with APF, while its clear composite pattern allows implementing more behaviors easily.

Finally, a generic design based on triggers and functors lets the final user implement complex scenarios of VR applications with the feeling of writing a script in C++. Applications developed on top of Imerse plug transparently into osgVR developed in the DREAM group (i.e., the research support development group of our Inria center). We are using osgVR to render OSG's scene graph in a distributed manner, since rendering clusters are available in an increasing number of installations. osgVR is a software layer developed by the DREAM research support group, ensuring synchronization and events/inputs distribution among a list of rendering slaves. These two libraries are available on GForge.

4.3. 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. 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, a previous 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 also been critical in establishing collaborations in the context of various grant proposals (EU and national).

4.4. 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.

VR4I Team

5. Software

5.1. OpenMASK: Open-Source platform for Virtual Reality

Participants: Alain Chauffaut [contact], Ronan Gaugne [contact], Georges Dumont, Thierry Duval, Marwan Badawi.

OPENMASK (Open Modular Animation and Simulation Kit) is a federative platform for research developments in the VR4i team. Technology transfer is a significant goal of our team so this platform is available as OpenSource software (http://www.openmask.org). OpenMASK is a C++ software platform for the development and execution of modular applications in the fields of animation, simulation and virtual reality. The main unit of modularity is the simulated object (OSO) which can be viewed as frequential or reactive motors. It can be used to describe the behavior or motion control of a virtual object as well as input devices control like haptic interfaces. Two OSO communicate with synchronous data flows or with asynchronous events. OpenMASK is well suited to develop applications in our new immersive room as ergonomics studies, including immersion, interaction, physic and haptic.

5.2. GVT : Generic Virtual Training

Participants: Bruno Arnaldi, Valérie Gouranton [contact], Florian Nouviale, Thomas Lopez, Andrés Saraos Luna.

The aim of GVT software is to offer personalized VR training sessions for industrial equipments. The most important features are the human and equipment security in the VR training (in opposition to the real training), the optimization of the learning process, the creation of dedicated scenarios, multiple hardware configurations: laptop computer, immersion room, distribution on network, etc.

The actual kernel of GVT platform is divided into two main elements that rely on innovative models we have proposed: LORA and STORM models.

- A Behavior Engine. The virtual world is composed of behavioral objects modelled with STORM (Simulation and Training Object-Relation Model).
- A Scenario Engine. This engine is used to determine the next steps of the procedure for a trainee, and its state evolves as the trainee achieves actions. The scenario is written in the LORA language (Language for Object-Relation Application).

A commercialized version of GVT, which includes a pedagogical engine developed in CERV laboratory, proposes training on individual procedures. A prototype is also available that enables users to train on collaborative procedures with one another or with virtual humans.

In the ANR Corvette 7.1.3 and in the FUI SIFORAS 7.1.1 , new features of GVT Software are proposed.

5.3. OpenViBE Software

Participants: Anatole Lécuyer [contact], Laurent Bonnet, Jozef Legény, Yann Renard.

OpenViBE is a free and open-source software devoted to the design, test and use of **Brain-Computer Interfaces**.

The OpenViBE platform consists of a set of software modules that can be integrated easily and efficiently to design BCI applications. Key features of the platform 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 LGPL-V2 licence, and it was officially released in June 2009. Since then, the OpenViBE software has already been downloaded more than 300 time, and it is used by numerous entities worldwide.

Our first international tutorial about OpenViBE was held at the International BCI Meeting in June 2010 (Monterey, US), with around 30 participants.

More information, downloads, tutorials, documentation, videos are available on OpenViBE website : http://openvibe.inria.fr

AXIS Project-Team

4. Software

4.1. Introduction

From its creation, AxIS has proposed new methods, approaches and software validated experimentally on various applications: Data Mining, Web usage Mining, Information Retrieval, Activity Modeling.
Some of our results are under process to be part of the FocusLab platform (CPER Télius 5.6) which is based on a Service oriented Architecture. The development process of the software part has started in 2011, finding ways to fund human ressources. Such a platform aims the community of Living Labs domain. In [70], we report the usage of the FocusLab platform (hardware and software components) inside various regional and european projects..

4.2. Data Mining

4.2.1. Classification and Clustering Methods

Participants: Marc Csernel, Yves Lechevallier [co-correspondant], Brigitte Trousse [co-correspondant].

We developed and maintained a collection of clustering and classification software, written in C++ and/or Java:

Supervised methods

- a Java library (Somlib) that provides efficient implementations of several SOM(Self-Organizing Map) variants [77], [76], [101], [100], [104], especially those that can handle dissimilarity data (available on Inria's Gforge server (public access) Somlib, developed by AxIS Rocquencourt and Brieuc Conan-Guez from Université de Metz.
- a functional Multi-Layer Perceptron library, called FNET, that implements in C++ supervised classification of functional data [96], [99], [98], [97] (developed by AxIS Rocquencourt).

Unsupervised methods : partitioning methods

- Two partitioning clustering methods on the dissimilarity tables issued from a collaboration between AxIS Rocquencourt team and Recife University, Brazil: CDis and CCClust [84]. Both are written in C++ and use the "Symbolic Object Language" (SOL) developed for SODAS. And one partitioning method on interval data (Div).
- Two standalone versions improved from SODAS modules, SCluster and DIVCLUS-T [74] (AxIS Rocquencourt).

Unsupervised methods : agglomerative methods

- a Java implementation of the 2-3 AHC (developed by AxIS Sophia Antipolis). The software is available as a Java applet which runs the hierarchies visualization toolbox called HCT for Hierarchical Clustering Toolbox (see [75]).

A Web interface developed in C++ and running on our Apache internal Web server .is available for the following methods: SCluster, Div, Cdis, CCClust.
Previous versions of the above software have been integrated in the SODAS 2 Software [95] which was the result of the european project ASSO [6] (2001-2004). SODAS 2 supports the analysis of multidimensional complex data (numerical and non numerical) coming from databases mainly in statistical offices and administration using Symbolic Data Analysis [71]. This software is registrated at APP (Agence de la Protection des Programmes). The latest executive version of the SODAS 2 software, with its user manual can be downloaded at http://www.info.fundp.ac.be/asso/sodaslink.htm [78], [85].

[6] ASSO: Analysis System of Symbolic Official data

As a 2012 result, a release of MND (Dynamic Clustering Method for Multi-Nominal data) algorithm based on previous AxIS research (2003) has been done (cf. section 5.6).

4.2.2. Extracting Sequential Patterns with Low Support

Participant: Brigitte Trousse [correspondant].

Two methods for extracting sequential patterns with low support have been developed by D. Tanasa in his thesis (see Chapter 3 in [103] for more details) in collaboration with F. Masseglia and B. Trousse :

- **Cluster & Divide**,
- and **Divide & Discover** [11].

These methods have been successfully applied from 2005 on various Web logs.

4.2.3. Mining Data Streams

Participants: Brigitte Trousse [correspondant], Mohamed Gaieb.

In Marascu's thesis (2009) [91], a collection of software have been developed for knowledge discovery and security in data streams. Three **clustering methods for mining sequential patterns (Java) in data streams** method have been developped in Java:

- SMDS compares the sequences to each others with a complexity of $O(n^2)$.
- SCDS is an improvement of SMDS, where the complexity is enhanced from $O(n^2)$ to $O(n.m)$ with n the number of navigations and m the number of clusters.
- ICDS is a modification of SCDS. The principle is to keep the clusters' centroids from one batch to another.

Such methods take batches of data in the format "Client-Date-Item" and provide clusters of sequences and their centroids in the form of an approximate sequential pattern calculated with an alignment technique.

In 2010 the Java code of one method called SCDS has been integrated in the MIDAS demonstrator and a C++ version has been implemented by F. Masseglia for the CRE contract with Orange Labs with the deliverability of a licence) with a visualisation module (in Java).

It has been tested on the following data:

- Orange mobile portal logs (100 million records, 3 months) in the context of Midas project (Java version) and the CRE (Orange C++ version)
- Inria Sophia Antipolis Web logs (4 million records, 1 year, Java version)
- Vehicle trajectories (Brinkhoff generator) in the context of MIDAS project (Java version).

In 2011, in the context of the ELLIOT contrcat [cf. Section 6.3.1.1), SCDS has been integrated as a Web service (Java version) in the first version of FocusLab platform (cf. section 5.6) in the ELLIOT context: a demonstration was made on San Rafaelle Hospital media use case at the first ELLIOT review at Brussels.

In 2012 we applied SCDS web service on data issued from co-creation step of two use cases in Logistics (BIBA) and Green Services (ICT Usage Lab). More data are needed to show the relevance of this method, it is planned in 2013 with the experimentation step of Green Services.

The three C++ codes done for the CRE (Orange Labs) have been depositi at APP.

4.3. Web Usage Mining

4.3.1. AWLH for Pre-processing Web Logs

Participants: Yves Lechevallier [co-correspondant], Brigitte Trousse [co-correspondant].

AWLH (AxIS Web Log House) for Web Usage Mining (WUM) is issued from AxISlogminersoftware which implements the mult-site log preprocessing methodology and extrcation of sequential pattern with low support developed by D. Tanasa in his thesis [15] for Web Usage Mining (WUM). In the context of the Eiffel project (2008-2009), we isolated and redesigned the core of AxISlogMiner preprocessing tool (we called it AWLH) composed of a set of tools for pre-processing web log files. The web log files are cleaned before to be used by data mining methods, as they contain many noisy entries (for example, robots requests). The data are stored within a database whose model has been improved.

So AWLH offers:

- Processing of several log files from several servers,
- Support of several input formats (CLF, ECLF, IIS, custom, ...),
- Incremental pre-processing,
- Java API to help integration of AWLH in external application.

An additionnal tool has been developped for capturing user actions in real time based on an open source project called "OpenSymphony ClickStream". An extension version of AWLH called **AWLH-Debate** has been developed for recording and structuring data issued from annotated documents inside discussion forums.

4.3.2. ATWUEDA for Analysing Evolving Web Usage Data

Participants: Yves Lechevallier [correspondant], Brigitte Trousse, Mohamed Gaieb, Yves Lechevallier [correspondant].

ATWUEDA for Web Usage Evolving Data Analysis [80] was developed by A. Da Silva in her thesis [79] under the supervision of Y. Lechevallier. This tool was developed in Java and uses the JRI library in order to allow the application of R which is a programming language and software environment for statistical computing functions in the Java environment.

ATWUEDA is able to read data from a cross table in a MySQL database. It splits the data according to the user specifications (in logical or temporal windows) and then applies the approach proposed in the Da Silva's thesis in order to detect changes in dynamic environment. The proposed approach characterizes the changes undergone by the usage groups (e.g. appearance, disappearance, fusion and split) at each timestamp. Graphics are generated for each analyzed window, exhibiting statistics that characterizes changing points over time.

Version 2.of ATWUEDA (september 2009) is available at Inria's gforce website.

In 2011 we have demonstrated the efficiency of ATWUEDA [82] by applying it on another real case study on condition monitoring data streams of an electric power plant provided by EDF.

ATWUEDA is used by Telecom Paris Tech and EDF [4].
This year we studied how to transform the code of ATWUEDA as a web service for the version 1.2 of FocusLab: in fact we gave up this objective, which would require more resource than we have.

4.4. Information Retrieval

*4.4.1. CBR*Tools for Managing and Reusing Past Experiences based on Historical Data*

Participant: Brigitte Trousse [correspondant].

CBR*Tools [87], [88] is an object-oriented framework [89], [86] for Case-Based Reasoning which is specified with the UMT notation (Rational Rose) and written in Java. It offers a set of abstract classes to model the main concepts necessary to develop applications integrating case-based reasoning techniques: case, case base, index, measurements of similarity, reasoning control. It also offers a set of concrete classes which implements many traditional methods (closest neighbors indexing, Kd-tree indexing, neuronal approach based indexing, standards similarities measurements). CBR*Tools currently contains more than 240 classes divided in two main categories: the core package for basic functionality and the time package for the specific management of the behavioral situations. The programming of a new application is done by specialization of existing classes, objects aggregation or by using the parameters of the existing classes.

CBR*Tools addresses application fields where the re-use of cases indexed by behavioral situations is required. The CBR*Tools framework was evaluated via the design and the implementation of several applications such as Broadway-Web, Educaid, BeCKB, Broadway-Predict, e-behaviour and Be-TRIP.

CBR*Tools is concerned by two past contracts: EPIA and MobiVIP.

CBR*Tools will be available for research, teaching and academic purpose via the FocusLab platform. The user manual can be downloaded at the URL: http://www-sop.inria.fr/axis/cbrtools/manual/.

See also the web page http://www-sop.inria.fr/axis/cbrtools/manual/.

*4.4.2. Broadway*Tools for Building Recommender Systems on the Web*

Participant: Brigitte Trousse [correspondant].

Broadway*Tools is a toolbox supporting the creation of adaptive recommendation systems on the Web or in a Internet/Intranet information system. The toolbox offers different servers, including a server that computes recommendations based on the observation of the user sessions and on the re-use of user groups' former sessions. A recommender system created with Broadway*tools observes navigations of various users and gather evaluations and annotations, to draw up a list of relevant recommendations (Web documents, keywords, etc).

Based on Jaczynski'thesis [87], different recommender systems have been developed for supporting Web browsing, but also browsing inside a Web-based information system or for query formulation in the context of a meta search engine.

4.5. Activity Modeling

4.5.1. K-MADe for Describing Human Operator or User Activities

Participant: Dominique Scapin [correspondant].

K-MADe tool (Kernel of Model for Human Activity Description Environment). The K-MADe is intended for people wishing to describe, analyze and formalize the activities of human operators, of users, in environments (computerized or not), in real or simulated situation; in the field, or in the laboratory. Although all kinds of profiles of people are possible, this environment is particularly intended for ergonomics and HCI (Human Computer Interaction) specialists. It has been developed through collaboration between ENSMA (LISI XSlaboratory) and Inria.
This year a new version v1.2 of K-MAD was released in december. Its history, documentation and tool are available at: http://kmade.sourceforge.net/index.php. This work follows up the findings from the work of Caffiau and al. [73].

DAHU Project-Team (section vide)

DREAM Project-Team

5. Software

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 [48] and [3].

The software is mainly implemented in Matlab. A standalone application is now avalaible. It uses the Mixmod toolbox [35] 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 presented in section 7.2.1 . 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 (see section 6.3.1). 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/.

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. 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 ManagerYourself 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.

EXMO Project-Team

5. Software

5.1. Alignment API

We have designed a format for expressing alignments in a uniform way [1]. The goal of this format is to be able 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, and has been defined by a DTD (Document Type Description for RDF/XML), an OWL ontology and an RDF Schema.

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, SWRL, OWL, C-OWL, SEKT mapping language, 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 wrapper for several ontology API;
- a parser for the format.

To instanciate 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

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 [12]. OntoSim provides a framework for designing various kinds of similarities. In particular, we differentiate similarities in the ontology space from those in the alignment space. The latter ones make use of 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.

OntoSim is written in Java and is available, under the LGPL licence, at http://ontosim.gforge.inria.fr.

In the continuation of our previous work, in 2012, we developed our work on evaluation of ontology matching and especially in running new experiments and generating new tests (§6.1.1). We introduced a new semantics for weighted correspondences (§6.1.2). We also continued our work on ontology matching for linking data (§6.2) and the use of the μ-calculus for evaluating RDF path queries (§6.3.1).

GRAPHIK Project-Team

5. Software

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 1.2, 2005–2010 GPL Licence). Currently, it supports Conceptual Graphs and import/export in RDFS.

Here are the major evolutions of the version delivered this year:

- Cogui now allows import/export in the Datalog+/- language thanks to a new Datalog+/- parser (see Sect. 5.4).
- Scripted rules were introduced. It is a new type of object that allows users to attach a script to a traditional rule in order to modify or control its behavior.
- A new interface ensures connectivity to a NoSQLdatabase (MongoDB).
- Large graphs can now be stored. In a near future, we will be able to perform queries on data too big to fit in central memory (see Sect. 5.2).

5.2. 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's thesis, and now of the PhD of Bruno Paiva Lima da Silva [34].

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 database used (*e.g.SPARQL* or *SQL*), or specifically designed algorithms (from a simple backtrack to a full constraint solver based upon *Choco* 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 (it is already used by other PhD students for their experiments), and will soon be ready to be distributed to a broader audience. Our generic approach will ease this dissemination to different research domains.

5.3. Kiabora

Participants: Swan Rocher [first year master internship], Michel LeclÃ¨re, Marie-Laure Mugnier.

http://www2.lirmm.fr/~mugnier/graphik/kiabora/index.html

Kiabora is a tool dedicated to the analysis of a set of existential rules. It can check if this set belongs to a known *decidable* class of rules, either directly or by means of its *Graph of Rule Dependencies (GRD)*. Kiabora analyzes the properties of the strongly connected components in the GRD, which allows to determine properties of the rule set with respect to decidability as well as the kind of paradigm (forward or backward chaining) ensuring decidability.

Besides, Kiabora also provides format conversion and rule decomposition services. It is written in Java.

5.4. DLGP

Participants: Jean-FranÃ§ois Baget, Michel LeclÃ¨re, Marie-Laure Mugnier, Alain Gutierrez, Swan Rocher [first year master internship], ClÃ©ment Sipieter [first year master internship].

http://www2.lirmm.fr/~mugnier/graphik/kiabora/downloads/datalog-plus_en.pdf

DLGP (for Datalog Plus) is a textual exchange format at once human-friendly, concise and easy to parse. This format can be seen as an extension of the commonly used format for plain Datalog. A file may contain four kinds of knowledge elements: facts, existential rules, negative constraints and conjunctive queries. This format will allow us to easily exchange data and ontologies with groups working on the equivalent Datalog+/- formalism, developped in Oxford.

A DLGP parser is now available.

For this section, participants are listed in alphabetical order.

MAIA Project-Team

5. Software

5.1. AA4MM

Participants: Vincent Chevrier [correspondant], Benjamin Camus.

Laurent Ciarletta (Madynes team, LORIA) is a collaborator and correspondant 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 [56] and written in Java. This year we added a new coupling between models to represent multi-level modeling, and rewrote a part of the core to ease coupling of simulator.

5.2. MASDYNE

Participant: Vincent Chevrier [correspondant].

This work was undertaken in a joint PhD Thesis between MAIA and Madynes Team. Laurent Ciarletta (Madynes 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.

5.3. FiatLux

FiatLux is a discrete dynamical systems simulator that allows the user to experiment with various models 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 2012, the main contributions were made by Olivier Bouré, who developed a lattice-gas cellular automata module.

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 Cartomatic 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 which as been protected by software copyrights (APP): 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.

MOSTRARE Project-Team

5. Software

5.1. FXP

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

Software Self-Assessment: A-3, SO-4, SM-3, EM-3, SDL-4

The FXP language is a temporal logic for a fragment of Forward XPath that is suitable for querying XML streams. The FXP library of the Mostrare project of Inria Lille provides rewriting tool that removes backward axis, a compiler of the FXP library to nested word automata and an efficient query answering algorithm for nested word automata on XML streams.
FXP is developed in the Inria transfer project QuiXProc in cooperation with Innovimax. Both a professional and a free version are available. The owner is Inria. A new release was published in October 2012.

See also the web page http://fxp.lille.inria.fr/.

- Version: FXP v1.1.0

5.2. QuixPath

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

Software Self-Assessment: A-3, SO-4, SM-3, EM-3, SDL-4

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. QuiXPath is developed in the Inria transfer project QuiXProc in cooperation with Innovimax. Both, a free open source and a professional version are available. The ownership of QuiXPath is shared between Inria and Innovimax. The main application of QuiXPath is its usage in QuiXProc, an professional implementation of the W3C pipeline language XProc own by Innovimax. A new release was published in October 2012.

See also the web page http://code.google.com/p/quixpath/.

- Version: QuixPath v1.1.0

5.3. VOLATA

Participant: Fabien Torre [correspondant].

Software Self-Assessment: A-2, SO-4, SM-2, EM-2, SDL-2

VOLATA provides several machine learning algorithms for attribute-value inference, grammatical inference and inductive logic programming.

See also the web page http://www.grappa.univ-lille3.fr/~torre/Recherche/Softwares/volata/.

- ACM: I.2.6

5.4. JProGraM

Participant: Antonino Freno [correspondant].

Software Self-Assessment: A-3, SO-3-up, SM-2, EM-3, SDL-4.

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.

See also the webpage http://researchers.lille.inria.fr/~freno/JProGraM.html.

OAK Team

5. Software

5.1. Amada

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

Contact: Jesús Camacho-Rodríguez (jesus.camacho-rodriguez@inria.fr))

Other contacts: Zoi Kaoudi (zoi.kaoudi@inria.fr), Ioana Manolescu (ioana.manolescu@inria.fr), Dario Colazzo (dario.colazzo@lri.fr), François Goasdoué (fg@lri.fr)

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

5.2. Nautilus Analyzer

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

Contact: Melanie Herschel (melanie.herschel@lri.fr)

Other contacts: n.a.

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

5.3. RDFViewS

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

Contact: Konstantinos Karanasos (konstantinos.karanasos@inria.fr)

Other contacts: François Goasdoué (fg@lri.fr), Julien Leblay (julien.leblay@inria.fr), and Ioana Manolescu (ioana.manolescu@inria.fr)

Presentation: a storage tuning wizard for RDF applications

5.4. ViP2P

Name: ViP2P (views in peer-to-peer, http://vip2p.saclay.inria.fr)

Contact: Ioana Manolescu (ioana.manolescu@inria.fr)

Other contacts: Jesús Camacho_Rodriguez (jesus.camacho-rodriguez@inria.fr), Asterios Katsifodimos (asterios.katsifodimos@inria.fr), Konstantinos Karanasos (konstantinos.karanasos@inria.fr)

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

5.5. XUpOp

Name: XUpOp (XML Update Optimization)

Contact: Dario Colazzo (colazzo@lri.fr)

Other contacts: Nicole Bidoit (bidoit@lri.fr), Marina Sahakian (Marina.Sahakyan@lri.fr), and Mohamed Amine Baazizi (baazizi@lri.fr)

Presentation: a general purpose type based optimizer for XML updates

5.6. XUpIn

Name: XUpIn (XML Update Independence)

Contact: Federico Ulliana (Federico.Ulliana@lri.fr)

Other contacts: Dario Colazzo (colazzo@lri.fr), Nicole Bidoit (bidoit@lri.fr)

Presentation: an XML query-update independence tester

5.7. XUpTe

Name: XUpTe (XML Update for Tempora documents)

Contact: Dario Colazzo (colazzo@lri.fr)

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

Presentation: a type-based optimizer for representing and updated XML temporal sata

5.8. XPUQ

Name: XPUQ (XML Partitioning for Updates and Queries)

Contact: Dario Colazzo (colazzo@lri.fr)

Other contacts: Nicole Bidoit (bidoit@lri.fr), Noor Malla (noorwm@hotmail.com)

Presentation: a static analyzer and partitioner for XML queries and updates

ORPAILLEUR Project-Team

5. Software

5.1. Generic Symbolic KDD Systems

5.1.1. The Coron Platform

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

data mining, frequent itemsets, frequent closed itemsets, frequent generators, association rule extraction, rare itemsets

The Coron platform [120], [102] 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, frequent closed itemsets, frequent generators. In this collection we can find APriori, Close, Pascal, Eclat, Charm, and, as well, original algorithms such as ZART, Snow, Touch, and Talky-G. The Coron-base component contains also algorithms for extracting rare itemsets and rare association rules, e.g. APriori-rare, MRG-EXP, ARIMA, and BTB. 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].

skyline, skycube algorithms

This program implements the algorithms described in a research paper published last year at VLDB 2010 [112]. 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].

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

It provides 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. A new version incorporating a graphic user interface was released and is now running on Windows systems.

CarottAge has been 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 [17].

5.2.2. The ARPEnTAge system

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

Hidden Markov Models, stochastic process

ARPEnTAge [1] (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 [106]. 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.

We model the spatial structure of the landscape by a Potts model with external field whose sites are LUS located in the parcels. The dynamics of these LUS are modeled by a temporal HMM2. This leads to the definition of a Potts model where the underlying mean field is approximated by a hierarchical hidden Markov model that processes a Hilbert-Peano fractal curve spanning the image.

Those stochastic models have been used to segment the landscape into patches, each of them being characterized by a temporal HMM2. The patch labels, together with the geographic coordinates, determine a clustered image of the landscape that can be coded within an ESRI shapefile. ARPEnTAge can locate in a 2-D territory time regularities and implements a Time-dominant approach in Geographic Information Systems.

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 have been used for mining decision rules in a territory holding environmental issues. They provide a way of visualizing the impact of farmers decision rules in the landscape and revealing new extra hidden decision rules [23].

5.2.3. GenExp-LandSiTes: KDD and simulation

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

simulation, Hidden Markov Models

In the framework of the project “Impact des OGM” initiated by the French ministry of research, we have developed a software called GenExp-LandSiTes for simulating bidimensional random landscapes, and then studying the dissemination of vegetable transgenes. The GenExp-LandSiTes system is linked to the CarottAge system, and is based on computational geometry and spatial statistics. The simulated landscapes are given as input for programs such as “Mapod-Maïs” or “GeneSys-Colza” for studying the transgene diffusion. Other landscape models based on tessellation methods are under studies. The last version of GenExp allows an interaction with R and deals with several geographical data formats.

[1] http://www.loria.fr/~jfmari/App/

This work is now part of an INRA research network about landscape modeling, PAYOTE, that gathers several research teams of agronomists, ecologists, statisticians, and computer scientists. Sébastien da Silva is preparing his PhD thesis within this framework and is conducted both by Claire Lavigne (DR in ecology, INRA Avignon) and Florence Le Ber [46], [40].

GenExp-LandSiTes was part of a survey about innovative tools for geographical information [74], [73]. This survey has been conducted within the GDR Magis and has been presented in a book both in French and in English.

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 made available on line (http://plateforme-mbi.loria.fr/intelligo/) to be used by members of the community for exploitation and 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 has been developed using rigorous software engineering practices and is used by professionals in both industry and academia. The KNIME environment 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.

A position of engineer ("Ingénieur Jeune Diplomé Inria") was granted to the Orpailleur team to develop some extra KNIME nodes for relational data mining using the ALEPH program (http://www.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.pl). The developed KNIME 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. An Inria APP procedure is currently pending.

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

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

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 steady-version of the software has been deposited 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. A poster was presented at the last JOBIM conference (Paris, June 2011). Recently, MODIM was used by colleagues from the LIFL for organizing data about non ribosomal peptide syntheses. Feedback from users led to extensions of the software. 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.

classification-based reasoning, case-based reasoning, edition and maintenance of knowledge, decision knowledge management, 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 of 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) has been 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.

The software 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 implemented thanks to a frequent close itemset extraction module of the Coron platform (see §5.1.1).

The Oncologik system is a collaborative editing tool aiming at facilitating the management of medical guidelines [49], [48]. Based on a semantic wiki, it allows the acquisition of formalized decision knowledge. A production version was released this year (http://www.oncologik.fr/). Oncologik also includes a graphical decision tree editor, KcatoS [61].

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].

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

Taaable 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 of knowledge engineering, namely 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, have been 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 since 2008 on line at http://taaable.fr, but is constantly evolving. This year, Taaable has been extended by two new features, both concerning knowledge acquisition.

The first feature uses closed itemsets for extracting adaptation knowledge in order to better adapt recipes. A first approach integrates a previous work about adaptation rule extraction [93] into a collaborative environment, in which humans and machines may now collaborate to better acquire adaptation rules [38]. This environment integrates also the results of a new work on knowledge extraction where specific cooking adaptation rules that can be applied to a single recipe, are generalized using close itemsets into generic adaptation rules, to make them usable on other recipes [60].

The second feature addresses the improvement of the formal representation of the preparation part of recipes, using a semi-automatic annotation process [59]. In Taaable, the procedural text describing the preparation is formalized in a graph, where cooking actions and ingredients, among others, are represented as vertexes, and semantic relations between those, shown as arcs. As the automatic annotation process that transforms, using natural language processing, a procedural text into a graph, produces incomplete annotation (disconnected graphs) or other annotation errors, a validating and correcting step is required. A specific graphical interface has been built to provide the users with a way to correct the graph representation of the cooking process, improving at the same time the quality of the knowledge about cooking procedures.

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

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

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 cased-based reasoning system, whose reasoning procedure is based on a domain ontology. This new system, called Tuuurbine, 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, Tuuurbine is built over semantic web standards that will ensure facilities for being plugged over data available on the web. The domain knowledge is considered to be represented in a RDF store, which could be additionally be interfaced with a semantic wiki, in order to benefit from the collaborative edition and management of the knowledge involved in the reasoning system (cases, ontology, adaptation rules). This development is support by an Inria ADT funding.

SMIS Project-Team

5. Software

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 [42] [32]. 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 [37]. Chip-Secured XML Access (C-SXA) was an XML-based access rights controller embedded in a smart card [38]. 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 details 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 [33] and a new version is registered each year. The underlying Flash-based indexing system has also been patented by Inria and Gemalto [43]. 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. 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 implementation. We have already discussed with hardware companies located in “île 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 [35]. It has been demonstrated at SIGMOD.

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

WAM Project-Team

5. Software

5.1. XML Reasoning Solver

Participants: Pierre Genevès, Nabil Layaïda, 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. 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 several milliseconds for comparison of XPath queries without tree types, to several seconds for queries under very large, 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 [5]. 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 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. Timesheets Library

Participants: Nicolas Hairon, Cécile Roisin.

The goal of the Timesheets library is to synchronize HTML5 content using declarative synchronization languages defined by W3C standards (namely SMIL Timing and Synchronization and SMIL Timesheets).

With the raise of HTML5 which natively supports continuous content (audio, video) there is a dramatic need for handling synchronization, animation and user interaction in an efficient and homogeneous way. As web browsers do not support SMIL, except for SVG Animation (which is based on the SMIL BasicAnimation module), multimedia web authoring remains difficult and relies on code-based, non-standard solutions.

Therefore we are developing a generic, cross-browser JavaScript implementation for scheduling the dynamic behavior of HTML5 content that can be described with declarative SMIL markup. Using a declarative language makes sense for the most common tasks, which currently require JavaScript programming:

- it is much easier for web authors and for web authoring tool developers;
- it is a much better way to achieve good accessibility and indexability;
- it is easier to maintain, since no specific JavaScript code is used.

This open source library is now deployed and used by external users. As far as we know, ENS Lyon was the first user: its site html5.ens-lyon contains several dozens of scientific conferences where the video capture of each conference is synchronized with the slides, a structured timeline and a table of contents. This web site was demonstrated in May at the WWW 2012 conference. University of Evry makes also a important use of the Timesheets library as a tool for teaching multimedia concepts at master level.

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 allows to take account of speed when walking, and 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 mobile. MAUDL is a cue-oriented interactive audio system, audio services being requested using named events and the systems response to each event being determined by the audio artist. The current version of the API supports iOS and further support for other mobile platforms (Android) is planned.

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 to superpose 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 POIs browsers, self-guided audio tours, or applications for following predefined routes.

5.4. Mixed Reality Browser (MRB)

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

The concept of Mixed Reality comes from the fact that the real/virtual dichotomy is not sharp, but interpolatively smooth over a virtuality continuum. Idealized notions of reality and virtuality can be thought of as endpoints on a continuum, an instance of the former approach corresponding for example to a see-through display with natural sounds, an instance of the latter to texture-mapped image-based rendering (panoramas) with synthetic sound objects.

Augmented Reality (AR) mode refers to all cases in which the auditory or visual display of an otherwise real environment is augmented by means of virtual sound or graphic objects. The converse case on the virtuality continuum is Augmented Virtuality (AV), where a virtual world, one that is generated primarily by computer, like with synthetic 3D graphic or synthetic panoramic, is being augmented with the audio-visual content of points of interest (POIs).

The Mixed Reality Browser (MRB) is a geolocalized web browser running on mobile devices. It uses standard and open XML formats for content authoring (HTML5, OSM and MAUDL) to allow anyone to create an augmented or virtual reality city tour that can be used with this application.

The introduction of mobile augmented reality browsers has forced a rethink on what kind of reality should be offered. Mobility induces a need for telepresence and simulation to free the user or the developer of the necessity to go every time in the real world. Mobility is the main reason behind the concept of the Mixed Reality Browser. By its intrinsic characteristics, MRB supports advance MR applications like mobile remote maintenance and assisted navigation.

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 sidewalk. GPS is 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 Unit (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 sidewalks 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.

WIMMICS Team

5. Software

5.1. Corese

Participants: Olivier Corby [correspondant], Fabien Gandon.

Corese [2] (COnceptual REsource Search Engine) is a Semantic Web Factory. It enables users to load and process RDFS schemas, RDF metadata and to query the base of annotations thus created, by using the SPARQL Query Language.

Corese implements RDF, RDFS and SPARQL 1.1 Query Language & Update. Furthermore, Corese query language integrates original features such as approximate search, SQL or XPath. Approximate search consists of searching the best approximate answers to a query according to the ontology types. Corese also integrates a SPARQL-based Rule Language for RDF.

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 benefited from Inria software development support (ADT) with two software engineers. 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 50 applications, 24 PhD Thesis and is used for education by several institutions. It has been used as a Semantic Web Factory in such projects as Ontorule, Palette, SevenPro and SeaLife european projects, in e-WOK Hub, Neurolog, ISICIL and Kolflow ANR projects, BioMarker and KmP projects, Semantic Web Import Plugin for Gephi visualization and ECCO ontology editor. The work on Corese was published in [95], [96], [97], [94], [1], [5], [3], [2], [4].

This year we completed the KGRAM SPARQL 1.1 Query & Update interpreter.

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

5.2. Semantic Web Import Plugin for Gephi visualization

Participants: Erwan Demairy, Fabien Gandon, Olivier Corby.

The SemanticWebImport [3] plugin is intended to allow the import of semantic data into Gephi open graph vizualisation platform. 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

[2] http://wimmics.inria.fr/corese
[3] http://wiki.gephi.org/index.php/SemanticWebImport

5.3. ISICIL

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

The ISICIL software platform is made of several software components:

- XUL (XML-based User interface Language) extensions for the Firefox browser to assist the technology watch and business intelligence tasks by collecting relevant metadata according to the navigation context of the user.
- An application server based on Tomcat publishes services using the REST protocol to process requests of the users' applications and in particular the navigation extensions.

This architecture is summarized in Figure 1 . Its major interest lies in the flexibility introduced by the loose coupling between REST services and navigators extensions or other applications.

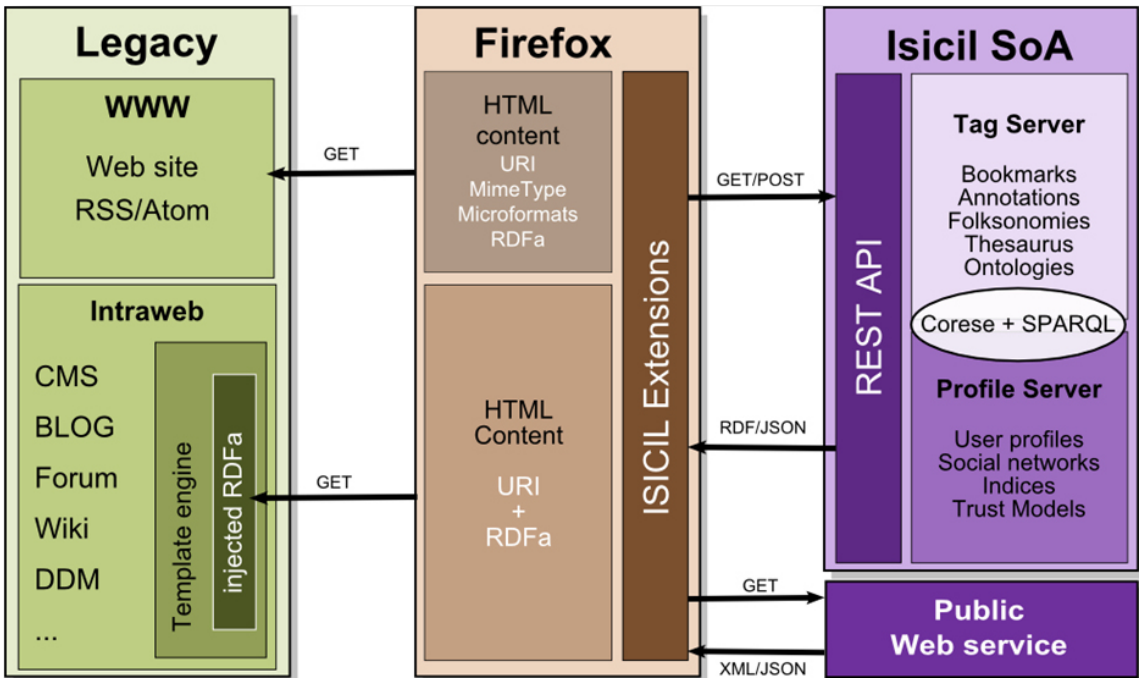


Figure 1. ISICIL Platform Architecture

In the context of the ISICIL ANR project, we have developped 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 SIOC [4], SCOT, and Newman's Tag Ontology[5] as shown in Figure 2 . 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 3 .

[4] http://sioc-project.org
[5] http://www.holygoat.co.uk/owl/redwood/0.1/tags

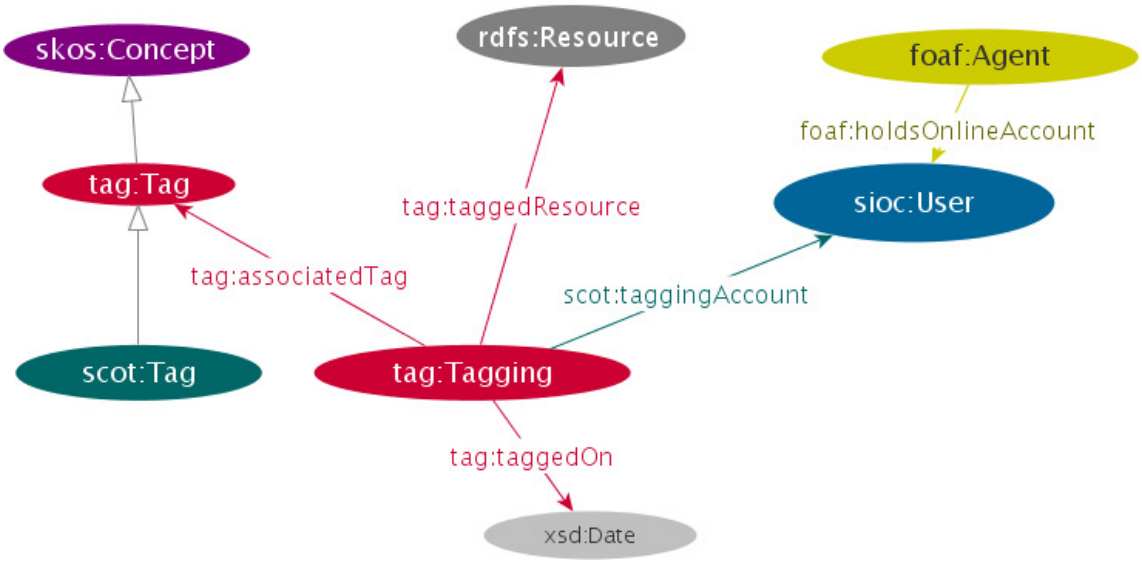


Figure 2. Model of tagging used in the Semantic Tag Server

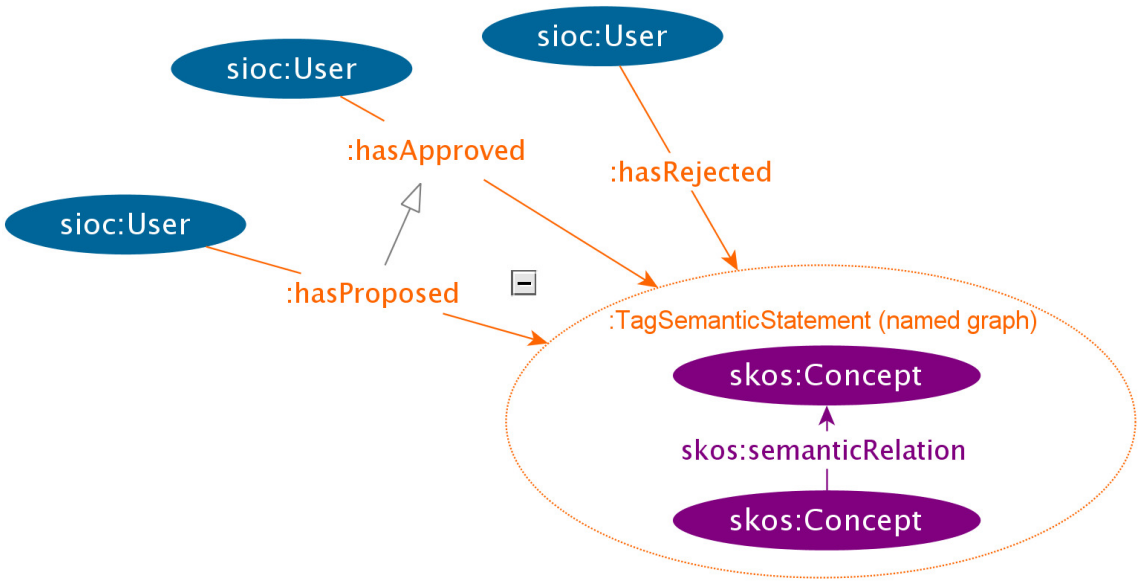


Figure 3. Folksonomy enrichment model

The functionnalities of this server can be divided into three categories:

1. Tagging : creating a tag; get tag suggestions based on the input characters; create a tagging, i.e. a link between a resource, a user, and a tag.
2. Computing: an external library (exported as a java jar file) has been developped to perform computations on the tagging data. Two types of computations have been implemented:
    1. Spelling Variant detection.
    2. Related tag detection based on the computation of the similarity between tags [101].
3. Managing Semantic relations between tags: get semantically related tags, reject or propose new semantic relations.

We developed a Firefox extension to help users navigate within a folksonomy and organize semantically the tags. The main idea behind this tool is to combine organization tasks with everyday tasks in the least intrusive way, that is to say, without forcing the user in any way, and by providing a user friendly graphical interface. This tool, developed using the XUL framework [6], is supported by the SRTag model and the Semantic Tag Server. Users are provided with search bar for navigating the folksonomy. When available, other tags are suggested and ordered according to their semantic relation with the searched tag (broader, narrower, related, spelling variant). Each suggestion can be either :

- clicked to search content tagged with this tag;
- rejected by clicking a checkbox;
- modified thanks to a drag-and-drop mechanism where a tag can be dropped in another category of semantic relation.

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

[6] http://developer.mozilla.org/en/XUL

ZENITH Project-Team

5. Software

5.1. WebSmatch (Web Schema Matching)

Participants: Zohra Bellahsène, Emmanuel Castanier, Rémi Coletta, Duy Hoa Ngo, 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. YAM++ ((not) Yet Another Matcher)

Participants: Zohra Bellahsène [contact], Duy Hoa Ngo, Konstantin Todorov.

URL: http://www2.lirmm.fr/~dngo/

YAM++ is a tool for discovering semantic correspondences between ontologies. YAM++ supports several matching strategies: machine learning; generic methods when learning data are not available; discovering alignment of ontologies represented in different languages. Furthermore, since this year YAM++ is able to deal with large scale ontology matching.

5.3. SON (Shared-data Overlay Network)

Participants: Ayoub Ait Lahcen, Fady Draidi, Esther Pacitti, Didier Parigot [contact], Patrick Valduriez, Guillaume Verger.

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.

5.4. P2Prec (P2P recommendation service)

Participants: Fady Draidi, Esther Pacitti [contact], Didier Parigot, Guillaume Verger.

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 the Data-Shared Overlay Network (SON) infrastructure which is the basis for the ANR DataRing project.

5.5. ProbDB (Probabilistic Database)

Participants: Reza Akbarinia [contact], Patrick Valduriez, Guillaume Verger.

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.

5.6. Pl@ntNet-Identify

Participants: Mathias Chouet, Hervé Goëau, Alexis Joly [contact].

URL: http://identify.plantnet-project.org

Pl@ntNet-Identify is a web application dedicated to the image-based identification of plants. It has been developed jointly by ZENITH, the AMAP UMR team (CIRAD) and the Inria team IMEDIA. It allows submitting one or several query pictures of a plant and browse the matching species in a large collection of social image data, i.e. plant images collected by the members of a social network. It also allows users to enrich the knowledge of the application by uploading their own pictures in the reference collection. Nowadays, the dataset includes more than 17K images posted by about 100 members of Telabotanica [1] social network. In 2012, about 5000 identification sessions have been recorded. The client side of the application is implemented in Javascript whereas the server side (visual search engine) is mostly implemented in C++.

5.7. Pl@ntNet-DataManager

Participants: Mathias Chouet [contact], Alexis Joly.

Pl@ntNet-DataManager 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. Pl@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 Pl@ntNet Project (CIRAD, INRA, Inria, IRD, Tela-Botanica) to manage taxonomical referentials, herbarium data and geolocated plant observations.

5.8. SnoopIm

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

URL: http://otmedia.lirmm.fr/OTmedia/

[1] http://www.tela-botanica.org/

SnoopIm is a content-based search engine allowing to discover and retrieve small visual patterns or objects in large collections of pictures (such as logos on clothes, road signs in the background, paintings on walls, etc.) and to derive statistics from them (frequency, visual cover, size variations, etc.). Query objects to be searched can be either selected from the indexed collection of photos, or selected from an external picture (by simply providing its URL). The web application allows online search of multiple users and has a cache feature to speed-up the processing of seen queries. It is implemented in Javascript on top of a C++ library developed in collaboration with INA [2]. The software is used at INA by archivists and sociologists in the context of the Transmedia Observatory project.

[2] http://www.ina-sup.com/

COPRIN Project-Team

5. Software

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.

5.2.2. Int4Sci : a Scilab interface for interval analysis

Participants: David Daney, Gilles Trombettoni, Bertrand Neveu.

In 2006, we have started the development of a Scilab interface to C++ Bias/Profil interval arithmetic package and to the library ALIAS. The first version of Int4Sci has been released in 2008 – see http://www-sop.inria.fr/coprin/logiciels/Int4Sci/ for linux, MacOS and Windows. A second version, compatible with Scilab 5.3 is in preparation . This interface provides an interval arithmetic, basic interval manipulation tools as well as the solving of linear interval systems. All functions are documented and a tutorial is available. Int4Sci is used in several universities for teaching the basis of interval analysis in place of using Rump's INTLAB for Matlab. We however lack the manpower to further enhance this software.

5.2.3. Mathematica Interface to Interval Analysis

Participants: Yves Papegay [correspondant], Jean-Pierre Merlet.

Since 2006, we have been implementing in Mathematica a high-level modular interface to the ALIAS library. Lack of manpower has slowed down this development.

E-MOTION Project-Team

4. Software

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 Hugr 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 sofware is implemented on the Lexus car. In 2012, a demand for deposing the GPU BOF software to the APP is in progress.

4.3. 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++

4.4. 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

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).

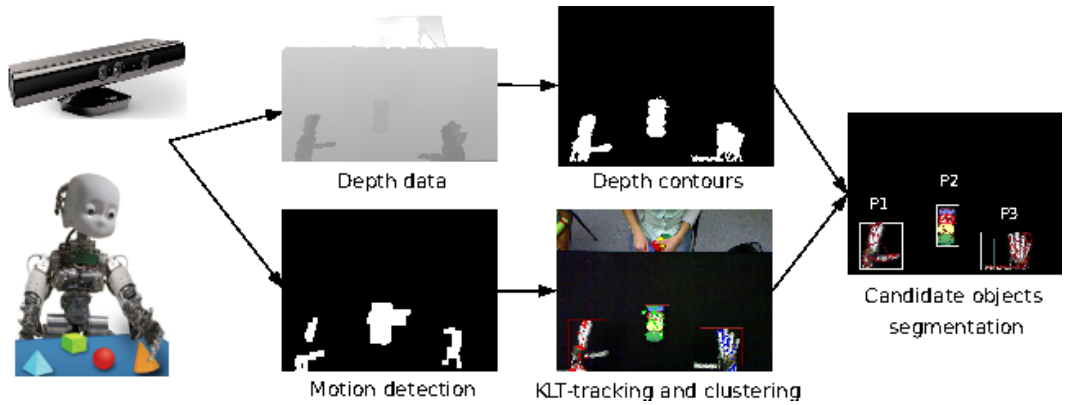


Figure 1. System Overview of the Incremental object discovery Software.

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 histrogram.

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 [39].

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 mulitple 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.

5.2. Datasets

5.2.1. Choreography dataset

Participants: Olivier Mangin [correspondant], Haylee Fogg.

This database contains choreography motions recorded through a kinect device. 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 [52] for studying learning techniques allowing to identify dictionaries of motion primitives, and is publicly available at https://flowers.inria.fr/choreography_database.html.

5.3. Learning algorithms

5.3.1. 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 2 .

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 [1]. Patrick Pilarski used RLPark for live demos on television (Breakfast Television Edmonton, CityTV, June 5th, 2012) and at TEDx Edmonton on Intelligent Artificial Limbs[2]. So far, RLPark has been used in more than a dozens of publications (see http://rlpark.github.com/publications.html for a list).

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 [3]. RLPark is licensed under the open source Eclipse Public License.

5.3.2. *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, Clement Moulin-Frier) and external collaborators (Jonas Buchli, Hwangbo Jemin of ETH Zurich). This code was used for the following publications: [63], [60], [62].

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

Participant: Alexander Gepperth [correspondant].

This simulation software implements the algorithms described in [24], [40]. 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.

[1] http://webdocs.cs.ualberta.ca/~sutton/Talks/Talks.html#sensorimotor
[2] http://www.youtube.com/watch?v=YPc-Ae7zqSo
[3] http://rlpark.github.com

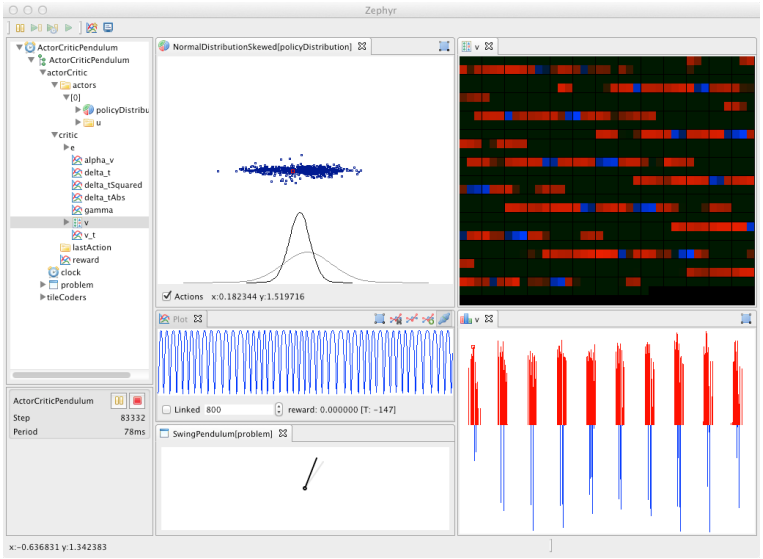


Figure 2. An example of an experiment in RLPark. Zephyr displays two views of a learned weight vector, an animation of the problem, the current policy distribution learned by the algorithm and the reward obtained by the algorithm. Videos are available at: http://rlpark.github.com.

5.3.4. 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 an 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 pyStreamPlaer 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 th replay and transmission across networks of distributed processes.

5.4. Software Platforms

5.4.1. Robust robotics manipulation - Object detection and tracking

Participants: Antoine Hoarau [ADT Engineer Since Nov. 2012], Freek 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. 3) 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 base their experiments. As the robot is not yet available at ENSTA, initial work focused on the robot's environment, meaning ROS and the M3 software (provided by Meka Robotics, based on both C++ and Python scripts) and on trying to implement a simple ball-catching demo : the idea is to throw a ball toward the robot which catch it (basic human-robot interaction, combining both perception and control). Different tracking algorithms are being tried for the ball, such as Camshift, Hough Circles + Kalman Filter, or more complex LineMod (all included in OpenCV) to finally estimate its trajectory for the robot to catch it. The M3 software provided by Meka Robotics contains a simulation environment that allows us to work without the robot hardware (cf. 4 .

Figure 3. The Meka robot plateform acquired by ENSTA ParisTech

5.4.2. ErgoRobot/Flowers Field Software

Participants: Jérôme Béchu [correspondant], Pierre-Yves Oudeyer, Pierre Rouanet, Olivier Mangin, Fabien Benureau, Mathieu Lapeyre.

In the context of its participation to the exhibition "Mathematics: A Beautiful Elsewhere" at Fondation Cartier pour l'Art Contemporain in Paris (19th October 2011 to 18th March 2012), the team has elaborated and experimented a robotic experimental set-up called "Ergo-Robots/FLOWERS Fields". This set-up is not only a way to share our scientific research on curiosity-driven learning, human-robot interaction and language acquisition with the general public, but, as described in the Results and Highlights section, attacks a very important technological challenge impacting the science of developmental robotics: How to design a robot learning experiment that can run continuously and autonomously for several months?

The global scenario for the robots in the installation/experiment is the following. In a big egg that has just opened, a tribe of young robotic creatures evolves and explores its environment, wreathed by a large zero that symbolizes the origin. Beyond their innate capabilities, they are outfitted with mechanisms that allow them to learn new skills and invent their own language. Endowed with artificial curiosity, they explore objects

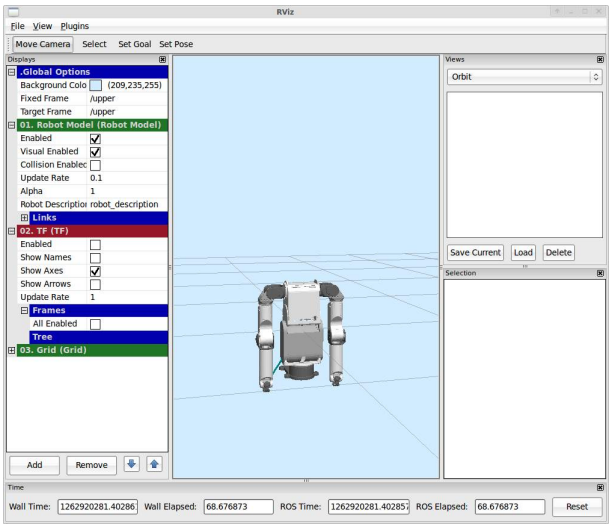


Figure 4. M3 simulation through ROS and Rviz

around them, as well as the effect their vocalizations produce on humans. Human, also curious to see what these creatures can do, react with their own gestures, creating a loop of interaction which progressively self-organizes into a new communication system established between man and ergo-robots.

We now outline the main elements of the software architectures underlying this experimental setup.

5.4.2.1. System components

The software architecture is organized to control the experiment at several levels, and in particular:

- **Scenes**: The organization of behavioural scenes, managing the behaviours that are allowed to each robot at particular times and in particular contexts;
- **Behaviours**: The individual behaviours of robots, also called stems, which are outlined in the next section;
- **stems**: The low-level actions and perceptin of robots while executing their behaviours, including motors control on the five physical stems, color and intensity of lights inside the stem head, production of sounds through speakers. Sensors are the kinect used to interact with visitors, and motor feedback capabilities.

In addition to that a video projector is used to display some artistic view of stem agents internal state.

5.4.2.2. Behaviours

A number of innate behaviours were designed and are used by the robots as elementary behaviours of more complex behaviours, including the three following learning behaviours.

The Naming Game is a behaviour played by stems two-by-two and based on computational models of how communities of language users can self-organize shared lexicons. In the naming game, stems interact with each other in a stylised interaction. Repeated interactions lead to the development of a common repertoire of words for naming objects. More precisely, object belong to meaning spaces. Two such spaces have been implemented for the exhibition. The first one is related to object spatial categorization and the second one is related to movement categorization. The object space contains stems, some hole in walls and the interaction zone. The movement space contains representations of small dances that stem can produce and reproduce.

Figure 5. Three importants concepts in ErgoRobots

Object Curiosity is a behaviour in controlling intrinsically motivated exploration of the physical environnement by the stems. A small wood object is present in the reachable physical environement of the stem, attached on the top of a spring so that it is guaranteed that it comes back to its original position. The stem uses a motor primitive to act on the object and motor feedback to detect movements of the object. The robot learns through active exploration what kind of parameters motor primitive will result in touching the object.

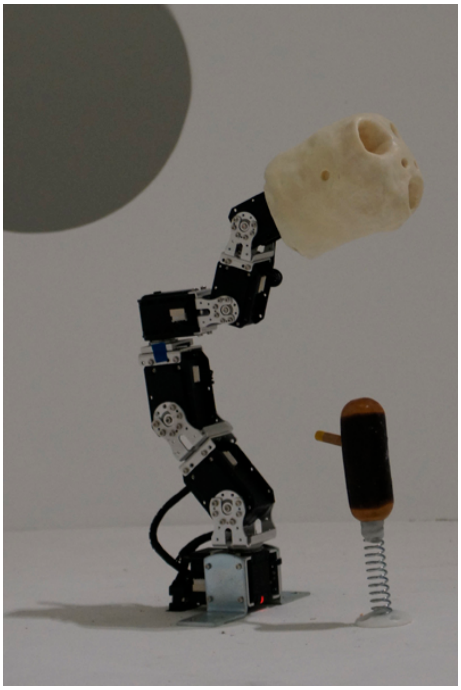


Figure 6. A Stem with the head designed by David Lynch and an Object

Birds Curiosity is a behaviour that drives robots to explore, through curiosity-driven learning, interaction with humans. One stem, generally the stem in the center, plays a sound, predicts the visitor reaction, look the interaction zone and wait the gesture of the visitor. To produce a sound the visitor have to make a gesture in space. In the next iterations, the robot chooses to produce sounds to human which produce most surprising

responses from the human (i.e. the robot is “interested” to explore sound interactions which are not easily predictable by itself).. As describe in the picture, the space is split in four. Each zone corresponding with a sound.

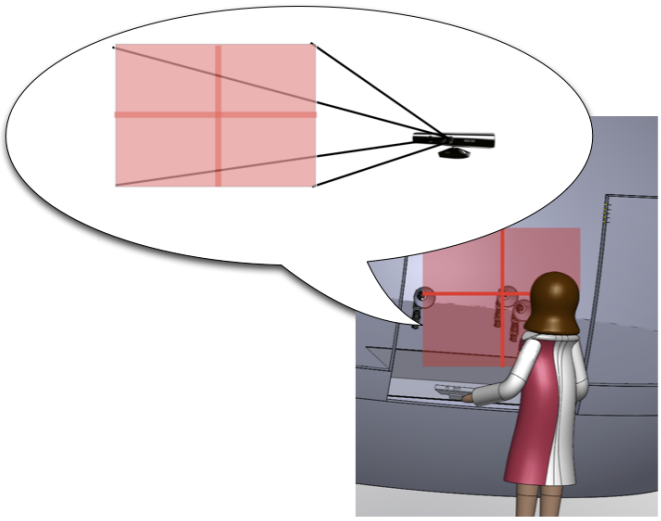


Figure 7. A virtual visitor interact with a virtual grid

5.4.2.3. Programming tools

The system is based on URBI and used some UObjects from UFlow. The most important part of the system is written in URBI script. Python and freenect [4] are used too.

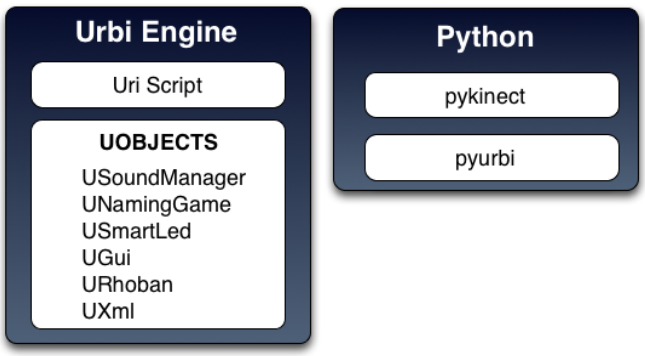


Figure 8. List of software used in ErgoRobots

[4] Kinect library

The system at the startup detects motors and lights. It create dynamically a list of Stem. A Stem is one robot with 6 motors as described in hardware part.

To interact with people, we used the freenect library to interface with the kinect, with a binding to python where detection and following of gestures is made.

For the display, we display an abstract rendering of the structure inside each ErgoRobot, using a python parser to read and parse log file from the ErgoRobot system, and the Bloom/Processing software to create and display the rendering. Currently, the system has three displays, one for the naming game, another one for birds curiosity and the last one for objects curiosity.

The sound system used the UObject USoundManager. It plays sounds when required by a behaviour, it also plays word sounds in Naming Game behaviour.

The Light system used Linkm technologies. In the head of each ErgoRobot we put two lights devices. Each light device is a RGB Light. We can control the intensity of each primary color through I2C control. To control lights we used LinkM USB Device. And finally we used an UObject dedicated to communicate with the USB Device.

5.4.2.4. Maintenance

A dedicate maintenance software is used to switch off, switch on the system. This software is written in Python (and Qt). The status of ErgoRobots is display on the graphical interface. Buttons are present too : Start, Stop, Reset and Take a video.

Recently we added a video system to have a visual feedback of motors usage and also to detect eventual problems. This is a screenshot of the application :

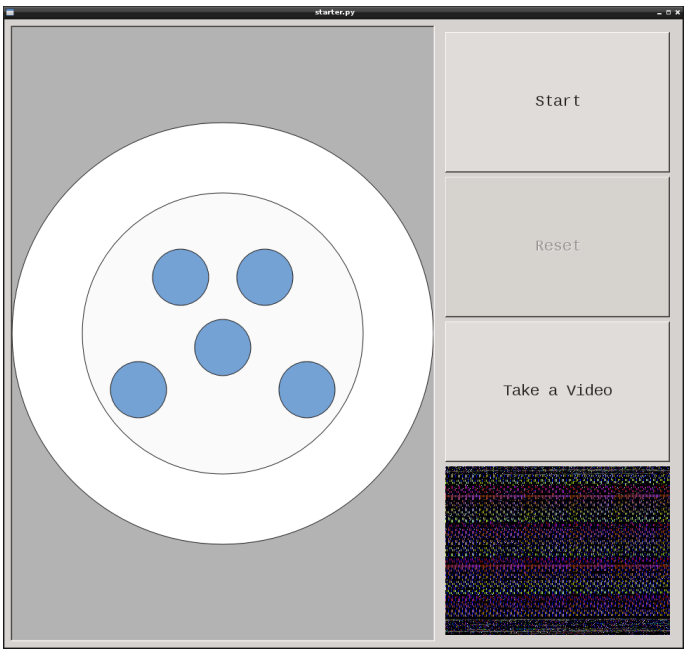


Figure 9. Maintenance Software for the ErgoRobots.

5.4.3. MonitorBoard - Complete solution for monitoring Rhoban Project robots

Participants: Paul Fudal [correspondant], Olivier Ly, Hugo Gimbert.

In collaboration with Rhoban Project/LaBRI/CNRS/Univ. Bordeaux I, the Flowers team took part in a project to exhibit robots at the International Exhibition in Yeosu - 2012 - South Korea (8 millions of visitors expected, from more than 100 countries). The installation consisted in three humanoids (one dancing, two playing on a spring) and five musicians (arms only) playing musical instruments (electric guitar, electric bass guitar, keytar, drums, DJ turntables). In order to increase the robustness of the robotic platform, a complete solution of software and hardware was build. The software solution aims to allow all robots to run safely during the whole exhibition (12 hours per days) and to provide an easy way to diagnose and identify potential electronic and mechanical failures. This software is able to monitor all robots at the same time, verify the health of each motors and each embedded systems. It is able to shutdown or reboot a robot if necessary using PowerSwitches (electric plugs controlled over network) and notify maintenance personal by email explaining the failure. All information is also logged for statistical use. This solution allows to monitor the whole platform without being present, and provides warning signs enabling preventive actions to be taken before an actual failures. It was entirely written in $C\#$ using Microsoft Visual Studio 2010 with .NET API and combined with the existing Rhoban Project API, extended and modified for this purpose. It also involved electric plugs controlled over a network connection.

5.4.4. Motor tracking system

Participants: Jérôme Béchu, Olivier Mangin [correspondant].

We developed a website interface to a database of motors used to build robots in the team. This system is designed for internal use in the team and was developed using the django web framework (https://www.djangoproject.com/) .

5.5. Visualization Tools

5.5.1. 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 10 .

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. Hardware

5.6.1. Poppy Platform

Participant: Matthieu Lapeyre [correspondant].

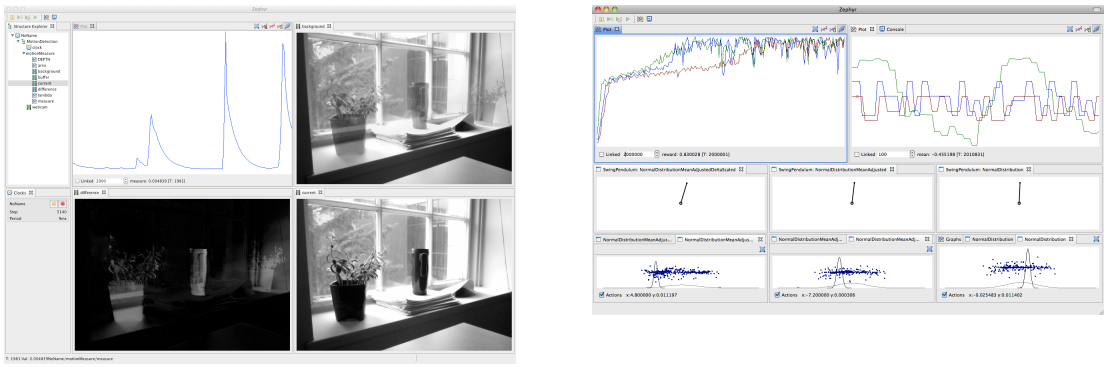


Figure 10. Left: Zephyr showing the different steps of a video processing pipeline in real-time. Right: Zephyr showing different data structure and variables of a reinforcement learning agent at different time scale. A video is available at: http://zephyrplugins.github.com.

5.6.1.1. Main goals :

No current platform (Nao [86], Darwin Op [87], Nimbro Op [117], 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

5.6.1.2. Overview :

Poppy platform (Figure 11) 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.

Poppy uses the bio-inspired trunk system introduced by Acroban [98]. 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.

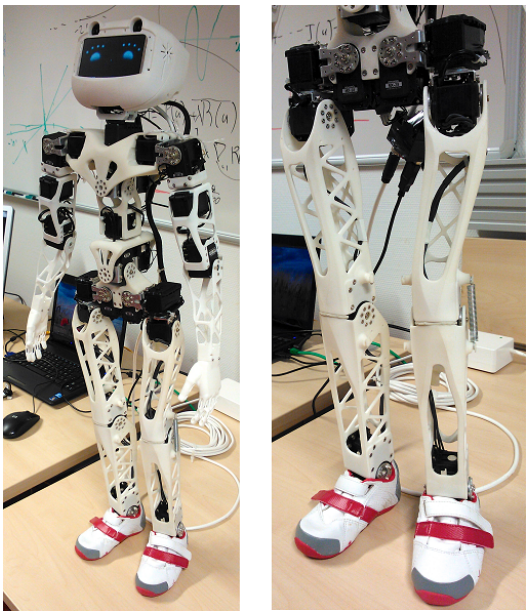


Figure 11. a. Global view of the Poppy platform. b. Zoom on legs design

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 12 .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 12 .b).

5.6.1.3. Future works :

In our work, we explore the combination of both a bio-inspired body and bio-inspired learning algorithms. We are currently working on experiments involving Poppy to perform skill learning. First we would like to succeed in achieving an effective postural balance using the articulated spine, the feet pressure sensors and the IMU. Then, we would like to perform experiments on the learning of biped walking using algorithms such as the ones described in [95] or [83]. We are expecting to clearly reduce the learning time needed and increase the quality of the learned tasks thanks to the bio-inspired morphology of Poppy.

We are also interested in social interactions with non-expert users. We would like to conduct user study to evaluate how playful physical interactions and emotions could improve learning in robotics. We think that the poppy platform could be very suitable for such studies.

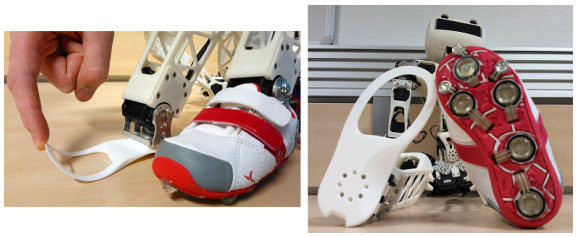


Figure 12. Poppy feet use actual children shoes combine with a compliant feet, toes (a.) and pressure sensors (b.)

5.6.2. Ergo-Robots/FLOWERS Fields: Towards Large-Scale Robot Learning Experiments in the Real World

Participants: Jerome Bechu, Fabien Benureau, Haylee Fogg, Paul Fudal, Hugo Gimbert, Matthieu Lapeyre, Olivier Ly, Olivier Mangin, Pierre Rouanet, Pierre-Yves Oudeyer.

In the context of its participation to the exhibition "Mathematics: A Beautiful Elsewhere" at Fondation Cartier pour l'Art Contemporain in Paris, starting from 19th October 2011 and to be held until 18th March 2012, the team, in collaboration with Labri/Univ. Bordeaux I, has elaborated and experimented a robotic experimental set-up called "Ergo-Robots/FLOWERS Fields" 13 . This set-up is not only a way to share our scientific investigations with the general public, but attacks a very important technological challenge impacting the science of developmental robotics: How to design a robot learning experiment that can run continuously and autonomously for several months? Indeed, developmental robotics takes life-long learning and development as one of its central objective and object of study, and thus shall require experimental setups that allow robots to run, learn and develop for extended periods of time. Yet, in practice, this has not been possible so far due to the unavailability of platforms adapted at the same time to learning, exploration, easy and versatile reconfiguration, and extended time of experimentation. Most experiments so far in the field have a duration ranging from a few minutes to a few hours. This is an important obstacle for the progress of developmental robotics, which would need experimental set-ups capable of running for several months. This is exactly the challenge explored by the Ergo-Robots installation, which we have approached by using new generations of affordable yet sophisticated and powerful off-the-shelf servomotors (RX Series from Robotis) combined with an adequately designed software and hardware architecture, as well as processes for streamlined maintenance. The experiment is now running for five months, six days a week, in a public exhibition which has strong constraints over periods of functioning and no continual presence of dedicated technicians/engineers on site. The experiment involved five robots, each with 6 degrees of freedoms, which are endowed with curiosity-driven learning mechanisms allowing them to explore and learn how to manipulate physical objects around them as well as to discover and explore vocal interactions with humans/the visitors. The robots are also playing language games allowing them to invent their own linguistic conventions. A battery of measures has been set up in order to study the evolution of the platform, with the aim of using the results (to be described in an article) as a reference for building future robot learning experiments on extended periods of time, both within the team and in the developmental robotics community. The system has been running during 5 months, 8 hours a day, with no major problems. During the two first months, the platform worked during 390h21mn, and was only stopped during 24h59mn (6 percent of time). After retuning the system based on what we learnt in the two first months, this performance was increased in the three last months: the platform worked for 618h23mn and was only stopped during 17h56mn (2.9 percent of time).
More information available at: http://flowers.inria.fr/ergo-robots.php and http://fondation.cartier.com/.

5.6.2.1. The Ergo-Robots Hardware Platform

Participants: Jerome Bechu [correspondant], Fabien Benureau, Haylee Fogg, Hugo Gimbert, Matthieu Lapeyre, Olivier Ly, Olivier Mangin, Pierre-Yves Oudeyer, Pierre Rouanet.

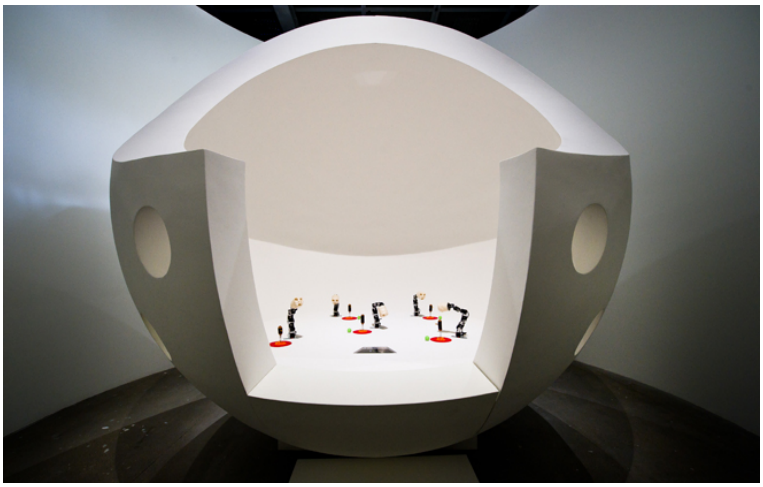


Figure 13. The Ergo-Robot experiment: robot learning experiment running continuously for 5 months at Fondation Cartier pour l'Art Contemporain, exhibition "Mathématiques: Un Dépaysement Soudain".

ErgoRobots 13 is a hardware platform for showcasing a number of curiosity and learning behaviours for the public to interact with. It was designed by the Flowers team in collaboration with Labri/Univ. Bordeaux I. The platform can also have future uses inside the lab for experiments that require more than one robot to complete. Although this system is entirely new this year, a very different previous version existed with the name FLOWERSField. It consists of five ErgoRobots, a control system, an interaction system, a display system, a sound system and a light system. There is an external system which monitors the ErgoRobots which contains a control system, a power system, a surveillance system and a metric capture system. The system has been running during 5 months, 8 hours a day, with no major problems. During the two first months, the platform worked during 390h21mn, and was only stopped during 24h59mn (6 percent of time). After retuning the system based on what we learnt in the two first months, this performance was increased in the three last months: the platform worked for 618h23mn and was only stopped during 17h56mn (2.9 percent of time).

The Ergo-Robot system: The robots themselves are each composed of six motors (see figure). Currently, the heads of the robots have been created in wax by David Lynch and the entire system is displayed at Fondation Cartier inside a large egg shaped orb as shown in the following diagram. The control system module contains both an MMNET1002 control board with an UART-RS485 breakout board which communicates with a ubuntu Linux PC via an ethernet cable. The mment board communicates with the motors, but all other ErgoRobot systems communicate with the PC directly. The sound system is currently externally provided and communicates with the PC. The light system is a series of two or three BlinkM RGB leds placed inside each ErgoRobot head that are controlled through two LinkM USB devices directly with the computer. A kinect placed in front of the system operates as the means for the public to interact with the platform and communicates directly through USB to the PC. The display system is currently an externally provided projector that projects visualisations of the field's current state behind the ErgoRobots.

The external system: This system allows anyone that is monitoring the system to externally control the ErgoRobots system. The PC with which the software control takes place is a Ubuntu Linux system which communicates with the ErgoRobot control system via an ethernet cable. The ErgoRobot harware system can be managed by an external power system which includes a 15.5V bench top power supply for the ErgoRobot motors, an external 12V plug in adapter for the mment board, an external 5V plug in adapter for the LED lights which are all controlled via an emergency stop button. The maintenance system can be located out of direct

view of the ErgoRobot field as it has a surveillance system: a kinect that can display the current state of the field. More surveillance is conducted through a metric capture system that communicates with the ErgoRobots to obtain various state values of the ErgoRobots through the motor sensors and other data. This surveillance is not entirely in place as of 2011 and will be implemented in early 2012.

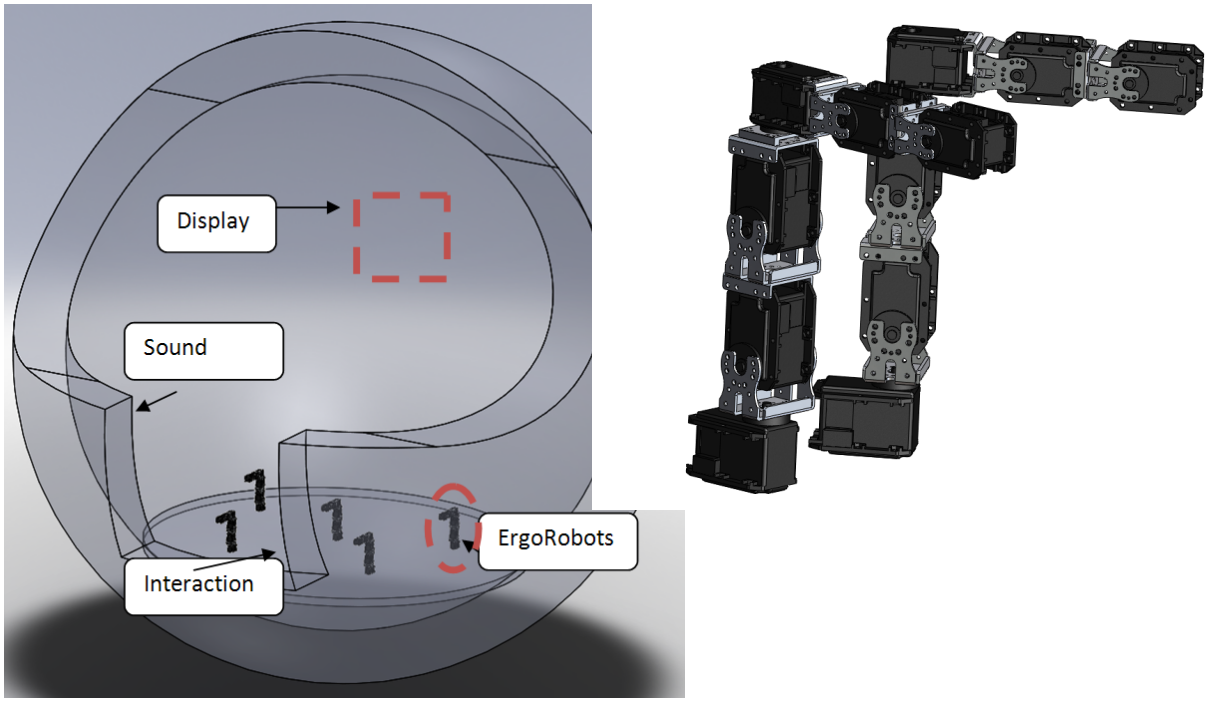


Figure 14. Ergo-Robots

5.6.2.2. Stem Platform for Affordances

Participant: Fabien Benureau [correspondant].

The Stem Platform for Affordances (figure 15 is a hardware platform that is intended for use in the lab for experiments. It features a 6 DOFs arm robot identical to the other robot stems present in the lab, and a physical platform intended for the interaction with objects. Our affordance experiments involves a lot of trials; there was the need for a platform that could reset itself after the robot interacted, as it is an assumption underlying our current algorithms. The stem platform provides exactly that, with the object position and orientation being reseted by the platform autonomously and in less than 10 seconds. This provides the potential to do more than 2000 independent interactions with an object over the course of 12 hours.

The platform also provides sensory capabilities, being able to track the position and orientation of the object at all time. On the hardware side, a camera is used. We investigated both a standard PSEye, that provides a high framerate (120Hz) with noise, and a high quality, firewire camera with professional optics, providing higher resolution, low noise at the expense of a low framerate (15Hz). The latter was kindly provided by the Potioc team. On the software side, computing the position is done by the open-source augmented reality library ARToolkitPlus. On the objects themselves, AR tags are placed.

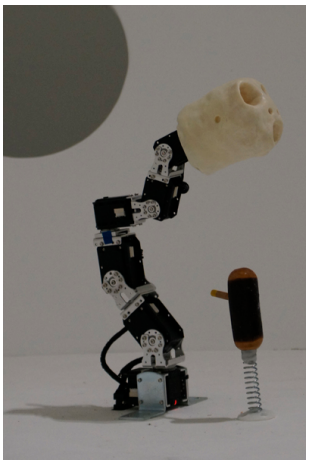


Figure 15. The Stem Platform

The platform is supported by a simulation that reproduce the setup in V-Rep. In order to be able to use the same algorithms for both the hardware and the simulation, a low-level interface was written for Pypot and V-Rep, using the work done by Paul Fudal on V-Rep Bridge.

The complete platform took roughly 3 weeks to make, with 3 additional weeks for the software. The team recently acquired material that would make possible to build a similar platform faster and in a more robust material (wood is used in the first platform). This platform, backed up by its simulation, will allow us to perform planned experiments in a reliable and statistically significant manner.

5.6.2.3. Humanoid Robot Torso

Participant: Haylee Fogg [correspondant].

The Humanoid Robot Torso is a hardware platform that is intended for use in the lab for either experiments or demonstrations 16 . It consists of a humanoid robot that contains just a torso, arms with shoulders and grippers, and head. It is entirely new this year, as a new design has been made, and a skeleton built with 3D printing technologies. The arms with the claws contain seven degrees of freedom (including 'grip'). The head consists of a smartphone for the face and an associated camera for the 'eyes' with the ability to move in two degrees (pitch and roll). The hardware is both robotis Dynamixel RX-28 and R-64 motors attached together with standard robotis frames and 3D printed limbs. A wiki has been built, documenting both the hardware and software platform.

5.6.2.4. NoFish platform

Participants: Mai Nguyen [correspondant], Paul Fudal [correspondant], Jérôme Béchu.

The NoFish platform is a hardware platform that is intended for use in the lab for experiments. It consists of an ErgoRobot with an attached fishing rod. The robot is fixed on a table and has in front of him a delimited area where to throw the fishing cap. This area is covered by a camera in order to track the fishing cap and to give its coordinates. The robot is managed by a software written using the Urbi framework. This program controls the robot using pre-programmed moves and also gives a way to uses the robot joint by joint. A second software written in C++ using OpenCV framework tracks the position of the fishing cap and sends the coordinates to the Urbi software controlling the robot. Finally, at the upper layer of the software architecture, MatLab is used to implement different learning algorithms. All MatLab code is able to receive informations from the Urbi part of the software (fishing cap coordinates, joints informations, etc) and also to send order to the robot (position joint by joint, preprogrammed moves, etc). To finish, and because the platform can run a learning algorithms

Figure 16. The Humanoid Robot Torso Platform

during a long time, an electric plug managed by the Urbi part of the software is added to the platform to shutdown the power if the robot is blocked or does not respond anymore.

IMARA Project-Team

5. Software

5.1. MELOSYM

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

MELOSYM is the acronym for "**M**odélisation de l'**E**nvironnement et **LO**calisation en temps réel pour un **SY**stème **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: 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.

- Version: V1

LAGADIC Project-Team

5. Software

5.1. ViSP: a visual servoing platform

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

Since 2005, we develop and release under the terms of the GPLv2 licence, ViSP, an open source library that 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].

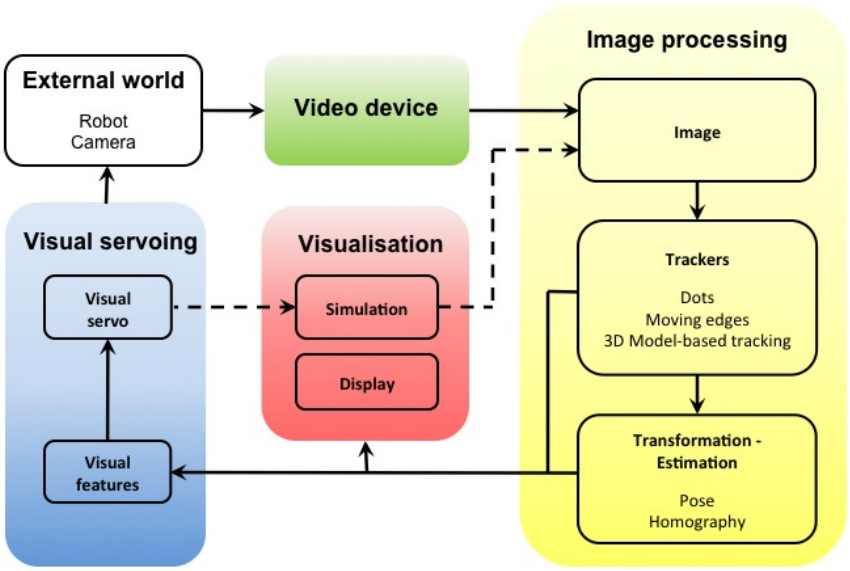


Figure 1. ViSP software architecture.

This year, we continued our efforts to improve the software and documentation quality. A new version available at http://www.irisa.fr/lagadic/visp/visp.html was released in July 2012. To ease ViSP installation, we provide also precompiled ViSP SDK including pre-built ViSP library and headers.

This last release under deposit to the APP ("Agence de Protection des Programmes") has been downloaded 887 times since its availability. It 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 a graduate course delivered at MIT, at IFMA Clermont-Ferrand and ESIR Rennes engineer schools. ViSP is now also part of "`vision_visp`" ROS stack (see http://www.ros.org/wiki/vision_visp) and ViSP 3D model-based tracker has been proposed by colleagues from Laas in Toulouse as a ROS package. This encouraged us to enhance "`vision_visp`" stack by proposing new ROS packages to calibrate intrinsic and extrinsic camera parameters, and a new 3D model-based tracker with automatic initialisation and reinitialisation after tracking loss (with help of specific textured patterns on the object).

5.2. DESlam

Participants: Patrick Rives [correspondant], Maxime Meilland.

The DESlam (Dense Egocentric Slam) software developed in collaboration with Andrew Comport from I3S in Sophia Antipolis was deposited 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. Development work: 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.

Two papers published by Lagadic in 2012 enclose results validated on this platform. Note that it is also opened to researcher from other labs. For example, this year an associate professor from LSIIT in Strasbourg did experiments on the Gantry robot.

5.4. Development work: Medical robotics platforms

Participants: Fabien Spindler [correspondant], Alexandre Krupa.

This tesbed 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 a six degrees of freedom Adept Viper S850 arm (see Fig. 3). This year we bought a new Adept Viper S650 arm to replace our eight year old Hippocrate medical arm designed by the Sinters company. 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 plan to exploit the two Viper robots for demonstrating needle insersion under ultrasound imaging to precisely guide the needle toward a target while optimizing its visibility (see Section 6.4.4).

Note that four papers published by Lagadic in 2012 enclose experimental results obtained with this platform.

5.5. Development work: Mobile robotics platforms

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

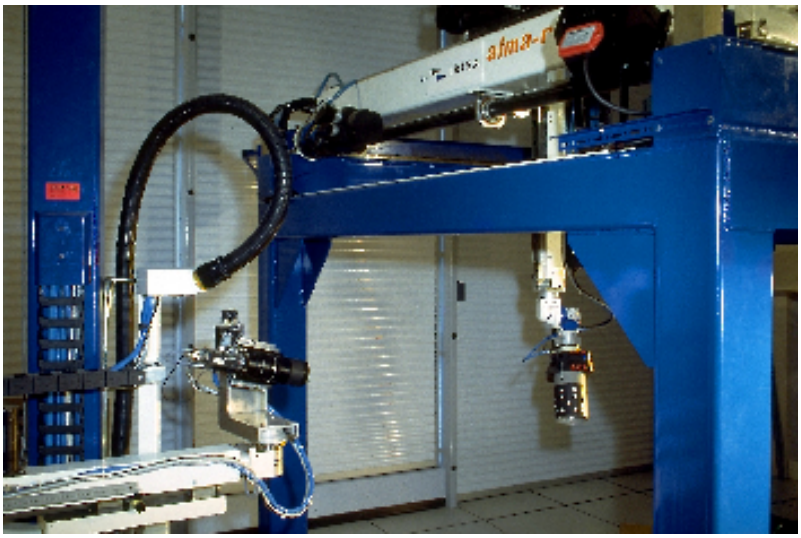


Figure 2. Lagadic robotics platforms for vision-based manipulation

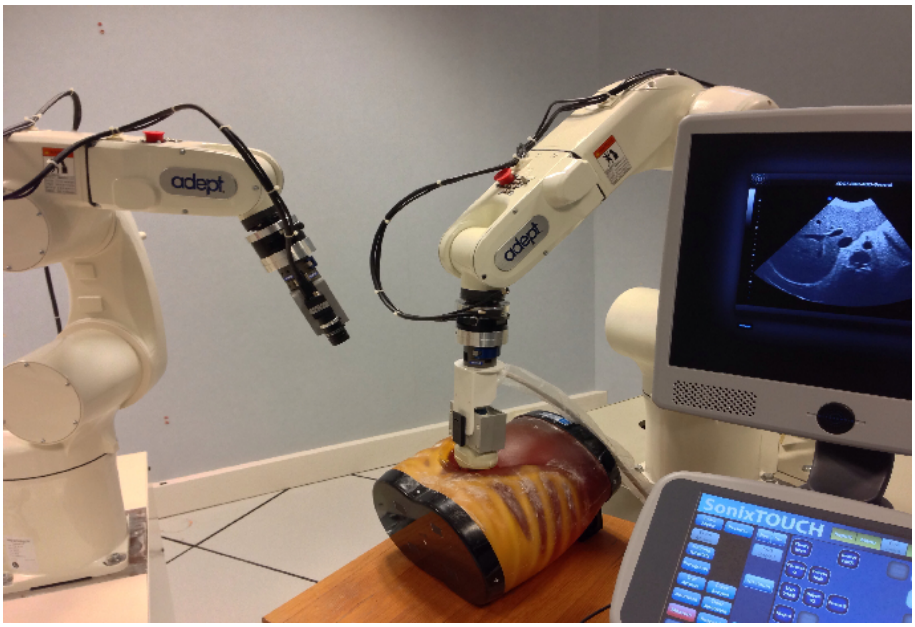


Figure 3. Lagadic medical robotics platforms. On the right Viper S850 robot arm equiped with a SonixTouch 3D ultrasound probe. On the left Viper S650 equipped with a tool changer that allows to attach a classical camera.

5.5.1. Indoors mobile robots

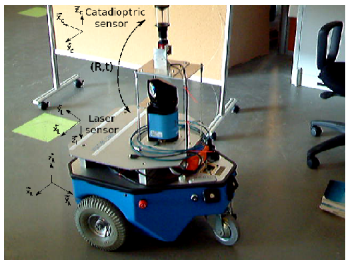
For fast prototyping of algorithms in perception, control and autonomous navigation, the team uses Hannibal in Sophia Antipolis, a cart-like platform built by Neobotix (see Fig. 4 .a), and a Pioneer 3DX from Adept in Rennes (see Fig. 4 .b) as well as a Robotino from Festo. These platforms are equipped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

Moreover, to validate the researches in personally assisted living topic (see 6.3.6), 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 three cameras to perform the required servoing for assisting handicaped people. Moreover, to ensure the direct security of the user, seven infrared proximity sensors have been installed all around the wheelchair.

5.5.2. Outdoors mobile robots

The team exploit 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.

Note that 5 papers published by Lagadic in 2012 enclose experimental results obtained with these mobile robotics platforms.



(a)

(b)

(c)

(d)

(e)

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

5.1. Software

5.1.1. Transfers

- The software MAD V2.0 was transfered to Galderma R&D in November 2012.
- The software Scombo v1.1 was transfered to Cutis laboratory (Galderma R& D, Sophia Antipolis) in May 2012, and to the French-Singaporean laboratory IPAL (Image and Pervasive Access Lab) in November 2012.

5.1.2. Deposits

- The software MAD (Melasma Automatic Detector) V2.0 was deposited with the APP in November 2012. A patent has also been deposited jointly by Galderma R& D and Inria during the same month. It deals with the melasma severity scoring from multi-spectral imaging.
- The software Scombo (Supervised Classifier of MultiBand Optical images) v1.1 was deposited with the APP in April 2012. It deals with the supervised classification of multiband optical images by using Markov random fields. It was developed with Aurélie Voisin, Vladimir Krylov and Josiane Zerubia.

IMEDIA2 Team

5. Software

5.1. IKONA/MAESTRO Software

Participants: Vera Bakic, Laurent Joyeux, Souheil Selmi.

IKONA is a generalist software dedicated to content-based visual information indexing and retrieval. It has been designed and implemented in our team during the last years [22]. Its main functionalities are the extraction, the management and the indexing of many state-of-the-art global and local visual features. It offers a wide range of interactive search and navigation methods including query-by-example, query-by-window, matching, relevance feedback, search results clustering or automatic annotation. It can manage several types of input data including images, videos and 3D models.

Based on a client/server architecture, it is easily deployable in any multimedia search engine or service. The communication between the two components is achieved through a proprietary network protocol. It is a set of commands the server understands and a set of answers it returns to the client. The communication protocol is extensible, i.e. it is easy to add new functionalities without disturbing the overall architecture. can be replaced by any new or existing protocol dealing with multimedia information retrieval.

The main processes are on the server side. They can be separated in two main categories:

- off-line processes: data analysis, features extraction and structuring
- on-line processes: answer the client requests

Several clients can communicate with the server. A good starting point for exploring the possibilities offere by IKONA is our web demo, available at https://www.rocq.inria.fr/cgi-bin/imedia/circario.cgi/bio_diversity?select_db=1. This CGI client is connected to a running server with several generalist and specific image databases, including more than 23,000 images. It features query by example searches, switch database functionality and relevance feedback for image category searches. The second client is a desktop application. It offers more functionalities. More screen-shots describing the visual searching capabilities of IKONA are available at https://www.rocq.inria.fr/cgi-bin/imedia/circario.cgi/demos.

IKONA is a pre-industrial prototype, with exploitation as a final objective. Currently, there does not exist a licensed competitor with the same range of functionalities. It exists several commercial softwares or systems exploiting technologies similar to some functionalities of IKONA but usually not the most advanced ones. We can for example cite the SDK developed by LTU company, the service proposed by AdVestigo company, etc. Many prototypes and demonstrators, industrial or academic, share some functionalities of IKONA but here again not the most advanced (e.g. Google Image Similarity Search Beta, IBM Muffin, etc.).

The main originality of IKONA is its **genericity** (in terms of visual features, metrics, input data, storage format, etc.), its **adaptivity** (to new visual features, new indexing structures or new search algorithms), its innovative **interactive** search functionalities (Local and Global Relevance Feedback, Local Search with Query Expansion, Search results clustering, etc.) and its **scalability** thanks to a generic indexing structure module than can support the integration of any new advances.

Current Users of IKONA include European and National Projects Participants through its integration in prototype multimedia systems, commercial companies through user trials (EXALEAD, INA, BELGA, AFP), General or Specific Public through Web demos (Pl@ntNet leaf identification demo).

IKONA software provides a high degree of visibility to IMEDIA2 scientific works through demos in commercial, scientific and general public events (notably in most Inria national showrooms). It is also the mainstay of several Multimedia Systems developed at the European level, in conjunction with many Leader European Companies and Research Centers.

LEAR Project-Team

5. Software

5.1. Face recognition

Participants: Guillaume Fortier [correspondant], Jakob Verbeek.

In a collaboration with Technosens (a start-up based in Grenoble) we are developing an efficient face recognition library. During 18 months Guillaume Fortier, financed by Inria's technology transfer program, had streamlined code developed by different former team members on various platforms. This encompasses detection of characteristic points on the face (eyes, nose, mouth), computing appearance features on these points, and learning metrics on the face descriptors that are useful for face verification (faces of the same person are close, faces of different people are far away). See http://lear.inrialpes.fr/~fortier/software.php.

5.2. 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 to reproduce the results from [19] 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.

5.3. 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 [6], [16] and in our submisssion to the Trecvid 2012 MED task [31]. 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.

5.4. Video descriptors

Participants: Dan Oneata, Cordelia Schmid [correspondant], Heng Wang.

We have developed and made on-line available software for video description based on dense trajectories and motion boundary histograms [28]. 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 scalability to large datasets. On the one hand we explored the effect of sub-sampling the video input both spatially and temporally, and evaluated the impact on the quality of the descriptors. On the other hand we avoid writing the raw MBH descriptors to disk, but rather aggregate them directly into a signature for the complete video using Fisher vectors, or bag-of-word descriptors. This allowed us to use these descriptors on the 4,000 hour video dataset of the TrecVid 2012 MED task.

MAGRIT Project-Team

5. Software

5.1. Software

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

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 proeminently 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.

5.1.2. Virtualization Gate

Vgate is an immersive environment that allows full-body immersion and interaction with virtual worlds. It is a joint initiative of computer scientists from computer vision, parallel computing and computer graphics from several research groups at Inria Grenoble Rhône-Alpes, and in collaboration with the company 4D View Solutions. The Morpheo team is leading this project.

Figure 1. Platforms: on the left the Grimage acquisition; on the right the vgate immersive environment.

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).

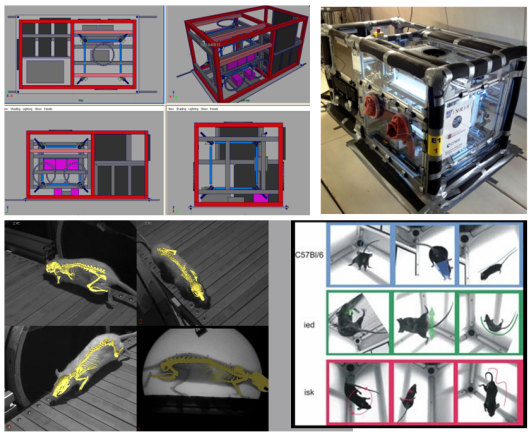


Figure 2. Ethomice: Experimental platform for video analysis of mice behavior.

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 iGlance, it is available as an open source software under the GNU LGP 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.

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

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 are 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 3. The mixed multi-camera system composed of four TOF-stereo sensor units.

5.2. Audiovisual robot head

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 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 (http://perception.inrialpes.fr/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 is being 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.

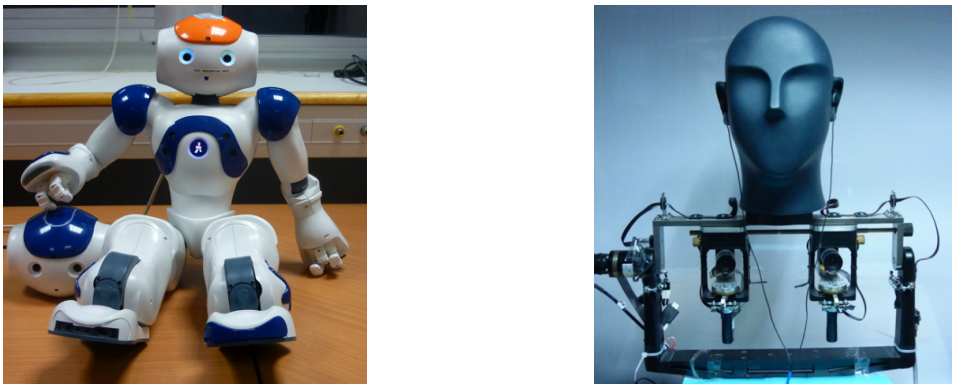


Figure 4. Left: The consumer humanoid robot NAO is equipped with a binocular-binaural head specially designed for human-humanoid interaction; Right: The binocular-binaural robot head POPEYE equipped with a four degrees of freedom stereo camera pair and with a dummy head.

PRIMA Project-Team

5. Software

5.1. OMiSCID Middleware for Distributed Multimodal Perception

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

Middleware, Distributed perceptual systems

OMiSCID is new 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.

5.2. 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 Versaille. 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.

Within CipeBus, Inria jas 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.

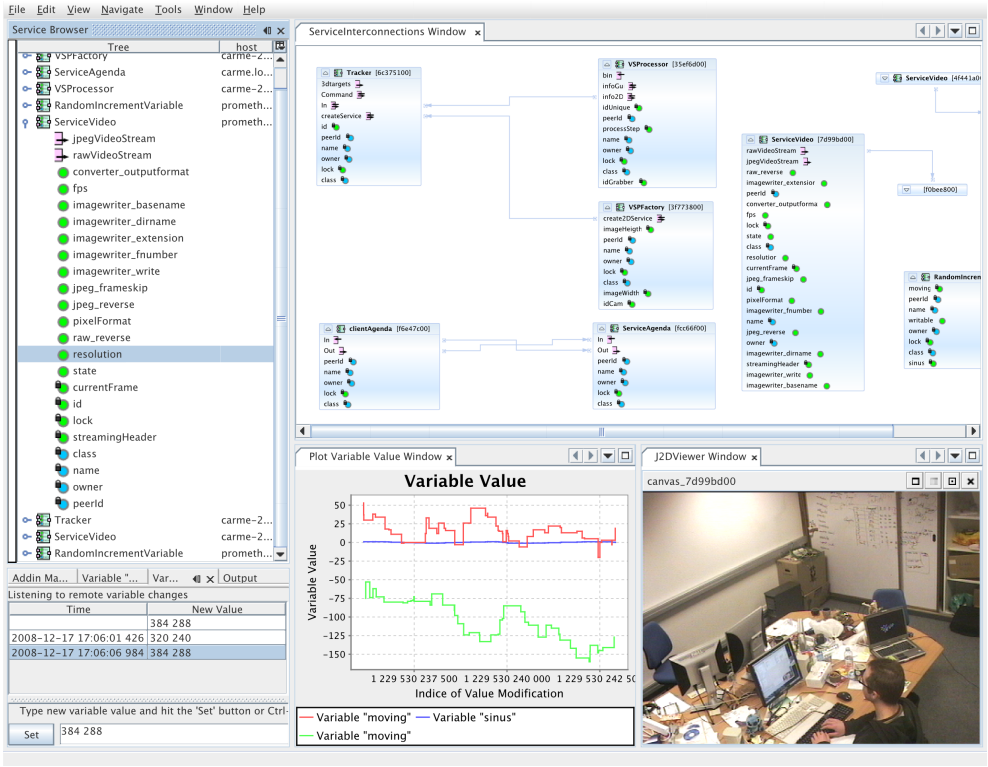


Figure 2. OMiSCID GUI showing a list of running services and some modules for service interconnections, variable plotting, live video stream display and variable control

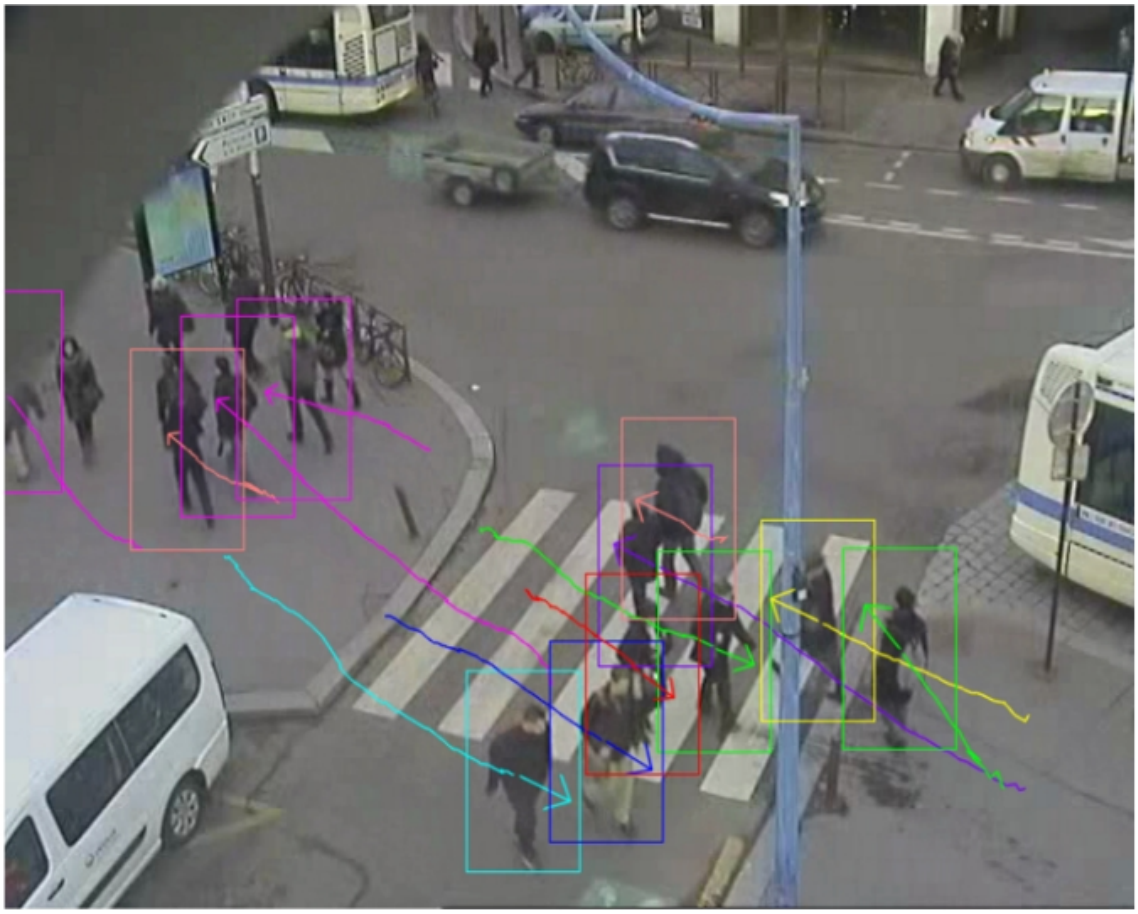


Figure 3. Cipebus: pedestrian tracking system.

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 pedertrians and public transportation while minimizing the delay imposed on private automobiles.

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

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

multimodal tracking of human activity

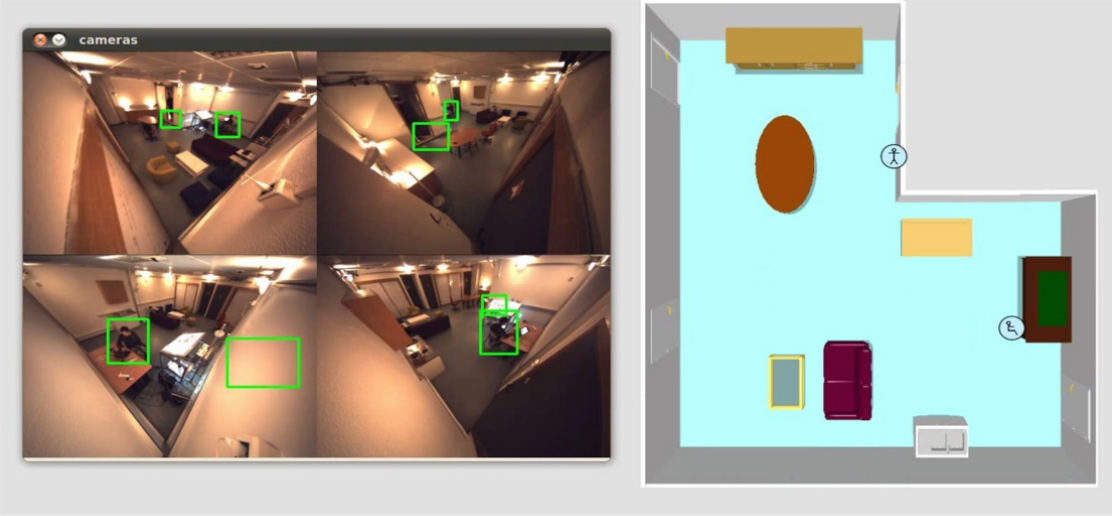


Figure 4. The 3D tracker integrates observations from multiple sensors

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.

5.4. 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

5.5. 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 AttentionEstimation.

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. Thesis displays can provide location aware access to information including maps and navigation guidance, information about local businesses and 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.

The software developed for this activity builds on face detections software that has recently been developed by Inria for the French OSEO project MinImage. MinImage was a five year, multi-million euro project to develop next generation technologies for integrated digital imaging devices to be used in cellphones, mobile and laptop computing devices, and digital cameras, that has begun in February of 2007. The project scope included research on new forms of retinas, integrated optics, image formation and embedded image processing. Inria was responsible for embedded algorithms for real time applications of computer vision.

Within MinImage, Inria developed embedded image analysis algorithms using image descriptors that are invariant to position, orientation and scale and robust to changes in viewing angle and illumination intensity. Inria proposed use of a simple hardware circuit to compute a scale invariant Gaussian pyramid as images acquired by the retina. Sums and differences of image samples from the pyramid provide invariant image descriptors that can be used for a wide variety of computer vision applications including detection, tracking and recognition of visual landmarks, physical objects, commercial logos, human bodies and human faces. Detection and tracking of human faces was selected as benchmark test case.

This work has been continued with support from EIT ICTlabs, to provide context information for interaction with large multi-touch interactive displays installed in public spaces.

Multitouch interactive displays are increasingly used in outdoor and public spaces. This objective of this task is to provide a visual observation system that can detect and count users of a multitouch display and to estimate information such as the gender, and age category of each user. us rendering the system sensitive to environmental context.

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, Activity 11547 : Pervasive Information interfaces and interaction. With Task 1207 of this activity we have constructed and released an "Attention Recognition Module". This software has been protected with an APP declaration.

An similar software was released in 2007 using face color rather than appearance. The system SuiviDeCiblesCouleur located individuals in a scene for video communications. FaceStabilsationSystem 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.

5.6. 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 the 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 additions, 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 [45] 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.

As part of Inria's contribution to ICT labs Action THWB Health and Wellbeing, Inria has participated in Activity 12100 "Affective Computing". In this activity we have provided a software system for detection, tracking of faces, and for visual measurement of Valence, Arousal and Dominance.

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 an mobile computing device using the Tegra 3 GPU.

SIROCCO Project-Team

5. Software

5.1. Oriented wavelet based image codec

Participant: Christine Guillemot [contact person].

This still image codec is based on oriented wavelet transforms developed in the team. The transform is based on wavelet lifting locally oriented according to multiresolution image geometry information. The lifting steps of a 1D wavelet are applied along a discrete set of local orientations defined on a quincunx sampling grid. To maximize energy compaction, the orientation minimizing the prediction error is chosen adaptively. This image codec outperforms JPEG-2000 for lossy compression. This software has been registered at the APP (Agence de Protection des Programmes) under the number IDDN.FR.001.260024.000.S.P.2008.000.21000.

5.2. M3DPlayer: 3D video player

Participant: Laurent Guillo [contact person].

A 3D player - named M3DPlayer - supporting rendering of a 3D scene and navigation within the scene has been developed. It integrates as a plug-in the 3D model-based video codec of the team. From a video sequence of a static scene viewed by a monocular moving camera, the 3D model-based video codec allows the automatic construction of a representation of a video sequence as a stream of textured 3D models. 3D models are extracted using stereovision and dense matching maps estimation techniques. A virtual sequence is reconstructed by projecting the textured 3D models on image planes. This representation enables 3D functionalities such as synthetic objects insertion, lightning modification, stereoscopic visualization or interactive navigation. The codec allows compression at very low bit-rates (16 to 256 kb/s in 25Hz CIF format) with a satisfactory visual quality. It also supports scalable coding of both geometry and texture information. The first version of the software was registered at the Agency for the Protection of Programmes (APP) under the number IDDN.FR.001.130017.000S.P.2003.000.41200.

A second version of the player has been registered at the APP (Agence de Protection des Programmes) under the number IDDN.FR.001.090023.000.S.P.2008.000.21000. In 2009-2010, we focused on improving the rendering engine, based on recent OpenGL extensions, to be able to render the viewed scenes on an auto-stereoscopic display with low-end graphic cards. In our case, auto-stereoscopic display requires the rendering of eight $1920x1200$ frames instead of just one for a standard display. This player is also used to render LDI (Layered Depth Images) and LDV (Layered Depth Videos) and to visualize 3D scenes on autostereoscopic displays taking multiple input views rendered from the LDI representation.

5.3. Depth maps extractor in mono-view (M3dAnalyzer2)

Participant: Laurent Guillo [contact person].

This software estimates depth maps from a video captured by a unique camera moving in a static 3D environment with Lambertian surfaces. These sequences are of interest to specialized applications such as augmented reality, remote-controlled robots operating in hazardous environments or remote exploration by drones. This software has been filed at the APP (Agence de Protection des Programmes) under the number IDDN.FR.001.110031.000.S.P.2010.000.31235.

5.4. Depth maps extractor in multi-view (MV2MVD)

Participant: Laurent Guillo [contact person].

This software estimates depth maps from multi-view videos, to provide Multi-View plus Depth (MVD) videos. MVD videos can be used to synthesize virtual views of the scene, or to render a different number of views than captured in the original video, for instance on an auto-stereoscopic display. This software produces depth maps of higher quality than those generated by the Depth Estimation Reference Software from the MPEG-3DV group, in terms of virtual views synthesis quality. This software has been filed at the APP (Agence de Protection des Programmes) under the number IDDN.FR.001.110034.000.S.P.2010.000.31235.

5.5. JPF-Joint Projection Filling

Participant: Fabien Racapé [contact person].

In the context of multi-view videos, this software generates virtual views of the scene from any viewpoint using a proposed method named Joint Projection Filling (JPF). The latter belongs to Depth-Image-Based Rendering (DIBR) methods, relying on warping equations, which project a reference view onto a virtual viewpoint. Each input view is defined by a "color" (or "texture") map and a "depth" map, which associates a depth value to each image pixel. The JPF method performs forward projection on depth map, using connectivity information to fill in disocclusions in a single step. Depth- based inpainting can then be used to fill in color disocclusions.

5.6. LDI builder

Participant: Fabien Racapé [contact person].

This software constructs a Layered Depth Image (LDI) representation of un-rectified Multi-View + Depth (MVD) sequences. The Incremental construction scheme reduces inter-layer correlation. The generated I-LDI is compatible with the M3DPlayer, permitting 3D visualisation and free viewpoint rendering of the 3D scene. The software also implements a virtual-view rendering technique which significantly reduces ghosting artefacts by eliminating untrusted texture boundaries detected in depth maps, as well as cracking artefacts thanks to an epipolar geometry aided inpainting method.

5.7. 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).

5.8. ADT PICOVIN-P

Participants: Laurent Guillo [contact person], Thomas Guionnet.

The ADT Picovin-P is a technological development action, which works closely with the project-team SIROCCO. This development structure is the follow-up of the ADT Picovin. It gives its support to the project-team to integrate new and relevant algorithms into the state-of-the-art video codec and to take part in standardization.

During this year, the ADT first pursued its developments on Intra prediction in the context of the standardization initiative referred to as High Efficiency Video Coding (HEVC) and led by the Joint ITU/MPEG Collaborative Team on Video Coding (JCT-VC). HEVC is implemented as a test model, the HEVC test Model (HM) in which the ADT tools have been integrated. We then followed the standardization activities within the Joint Collaborative Team on 3D Video Coding Extension (JCT-3V). JCT-VC and JCT-3V have been both created by the ITU-T Study Group 16 (VCEG) and the ISO/IEC JTC 1/SC 29/WG 11 (MPEG). While JCT-VC aims at developing a new generation 2D video coding standard, JCT-3V aims at developing 3D extensions for video codecs, which are AVC (ATM) or HEVC (HTM) based.

As part of JCT-3V, we submitted several proposals related to the handling of the merge list of predictor candidates. They were about the re-ordering of the candidates in the list and about the addition of new candidates. Two of them have been accepted in the dedicated core experiment (CE) during the 1st JCT-3V meeting which was held in Stockholm in July 2012. An improved version related to the addition of candidates has been accepted in a CE during the 2nd JCT-3V meeting in Shanghai in October 2012. It will be integrated in the coming test model (HTM) and evaluated during the next meeting in Geneva in January 2013. During 2012, the ADT also took part in cross checks which aims at evaluating and testing tools studied in core experiments. As part of cross checks related to JCT-VC or JCT-3V the ADT ran 9 tests jointly with companies such as Canon, Huawei, HiSilicon and Nokia. The ADT Picovin-P started in October 2011 and lasted one year. During this year, one permanent engineer from the SED Rennes (development and experimentation department of Inria Rennes) and one senior engineer specialized in video compression are involved in the ADT. It is supported by the technological development department of Inria.

STARS Team

5. Software

5.1. SUP

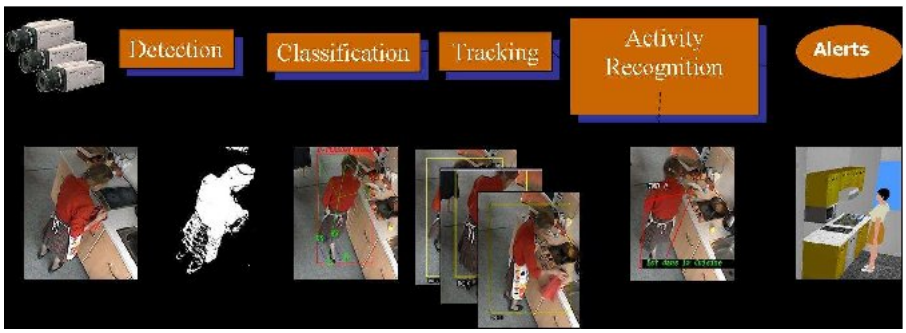


Figure 5. Tasks of the Scene Understanding Platform (SUP).

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).

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 ViSEvAl 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. ViSEvAl 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/ViSEvAl_Description.html

5.3. Clem

The *Clem Toolkit* [63](see Figure 9) is a set of tools devoted to design, simulate, verify and generate code for LE [19] [77] programs. LE is a synchronous language supporting a modular compilation. It also supports automata possibly designed with a dedicated graphical editor.

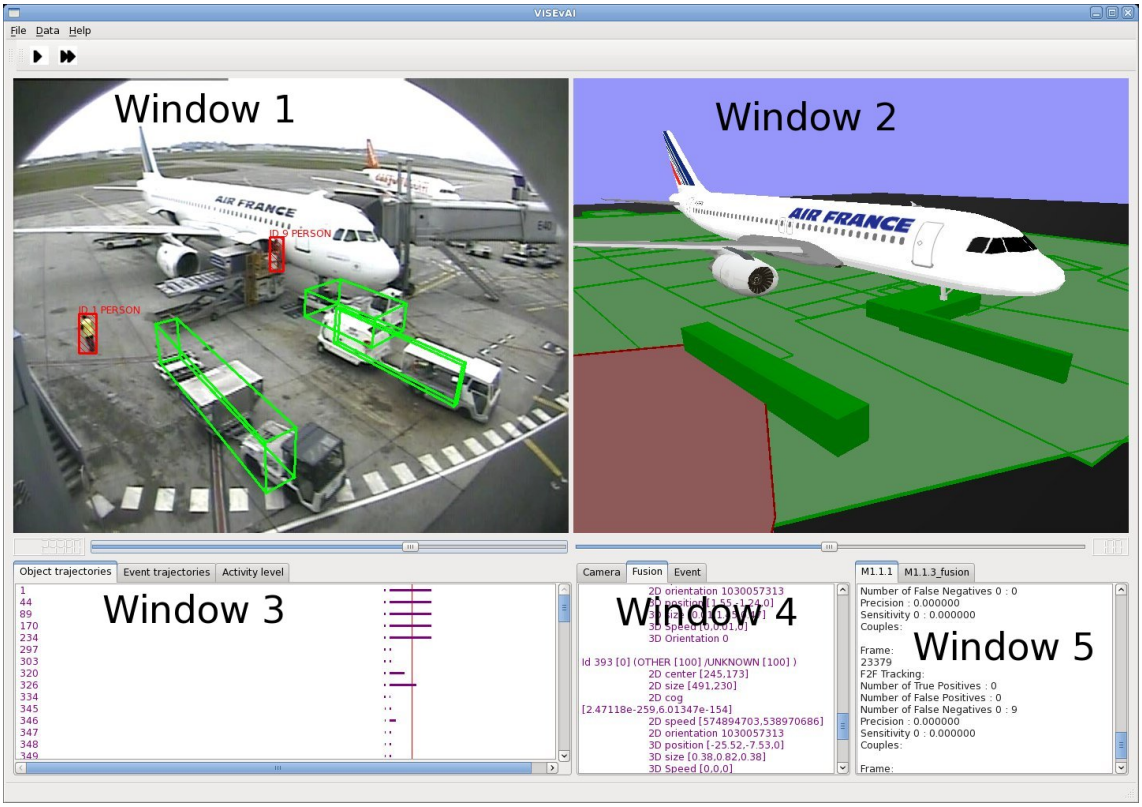


Figure 6. GUI of the ViSEvAl software

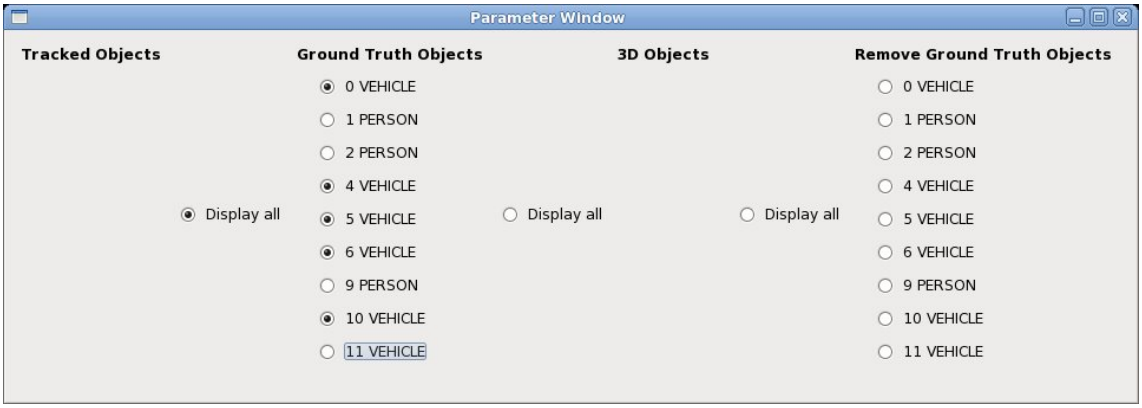


Figure 7. The object window enables users to choose the object to display

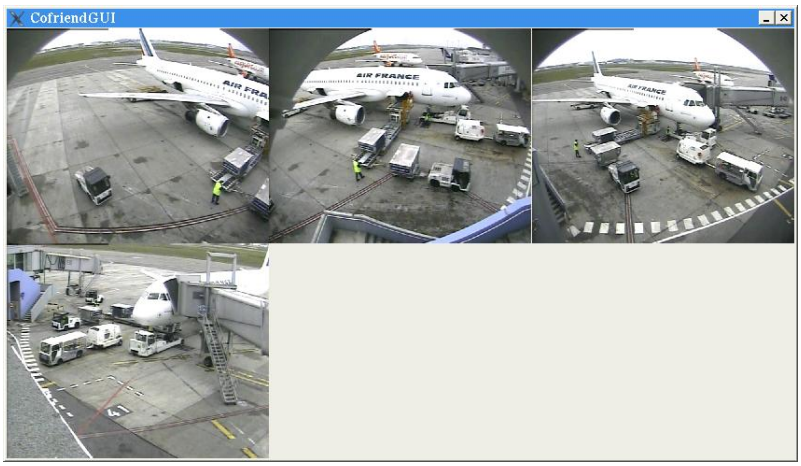


Figure 8. The multi-view window

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 [61] in order to perform validation of program behaviors.

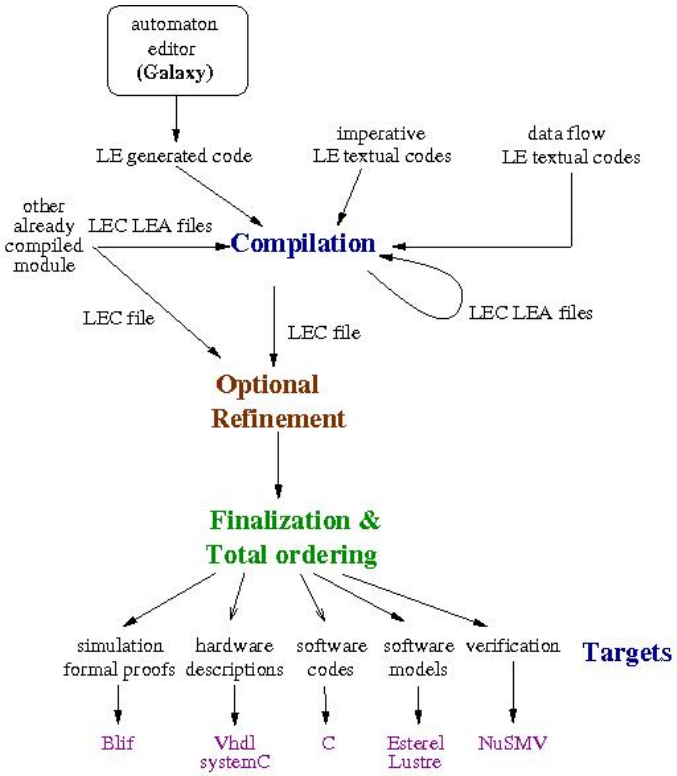


Figure 9. The Clem Toolkit

TEXMEX Project-Team

5. Software

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. 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 / indexation engine.

First APP deposit: IDDN.FR.001.4200010.000.S.P.2012.000.20900.

5.1.1.2. Peyote

Participants: Sébastien Campion, 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 and was used for the Mediaeval Placing task [68] and the Trecvid Instance Search task.

First APP deposit: IDDN.FR.001.4200008.000.S.P.2012.000.20900.

5.1.1.3. Watermarking Effective Key Length Evaluation

Participant: Teddy Furon [correspondent].

This software was developed in collaboration with Patrick BAS (CNRS, Ecole Centrale de Lille)

Weckle is a software suite in Matlab and R for the numerical evaluation of the effective key length of watermarking schemes based on Spread Spectrum, a concept which was proposed in [22], [23].

5.1.2. Most active software started before 2012

5.1.2.1. Babaz

Participants: Jonathan Delhumeau, Guillaume Gravier, Hervé Jégou [correspondent].

Babaz (http://babaz.gforge.inria.fr/) is a audio database management system with an audio-based search function, which is intended for audio-based search in video archives. First APP deposit: IDDN.FR.001.010006.000.S.P.2012.000.10000. It is licensed under the terms of the GNU General Public License v3.0.

5.1.2.2. Bigimbaz

Participant: Hervé Jégou [correspondent].

Bigimbaz is a platform originally developed in the LEAR project-team, and now co-maintained by TEXMEX. It integrates several contributions on image description and large-scale indexing: detectors, descriptors, retrieval using bag-of-words and inverted files, and geometric verification.

5.1.2.3. BonzaiBoost

Participant: Christian Raymond [correspondent].

The software homepage is 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

5.1.2.4. 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.

5.1.2.5. Nero

Participant: Sébastien Campion [correspondent].

The service is available at https://nero.irisa.fr.

NERO is an online Named Entities Recognition system. It is implemented within a web service that allows other member of the community to evaluate our results online without any client side setup. An HTTP Rest API, Shell and Python client are provided. The protocol used is HTTPS to secure the transactions between the user and the server. A user account is needed, which allow a fine monitoring. Usage are also limited to 100 thousand characters per account.

5.1.2.6. SURVA

Participants: Sébastien Campion [correspondent], Jonathan Delhumeau.

Speed Up Robust Video Alignement enables to quickly and efficiently synchronize the same video with two coding and quality formats (i.e. without the same number of frame). First APP deposit: IDDN.FR.001.420009.000.S.P.2012.000.20900.

5.1.2.7. PimPy

Participant: Sébastien Campion [correspondent].

PIMPY provides a convenient and high level API to manage common multimedia indexing tasks. It includes severals features. It is used, in particular

- to retrieve video features, such as histogram, binarized DCT descriptor, SIFT, SURF, etc ;
- to detect video cuts and dissolve (GoodShotDetector) ;
- for fast video frame access (pyffas) ;
- for raw frame extraction, or video segment extraction and re-encoding ;
- to search a video segment in another video (content based retrieval) ;
- to perform scene clustering.

First APP deposit: IDDN.FR.001.260038.000.S.P.2011.000.40000

5.1.2.8. Pqcodes

Participant: Hervé Jégou [correspondent].

This software is jointly maintained by Matthijs Douze, from Inria Grenoble.

Pqcodes is a library which implements the approximate k nearest neighbor search method of [83] 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.

5.1.2.9. Yael

Participant: Hervé Jégou [correspondent].

This software is jointly maintained by Matthijs Douze, from Inria Grenoble.

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 1000 times in 2012.

The current version registered at APP is IDDN.FR.001.220014.001.S.P.2010.000.10000.

5.1.2.10. IRISA News Topic Segmenter (irints)

Participants: Guillaume Gravier [correspondent], Camille Guinaudeau, 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, implements a technique to describe the images based on color.
- I-DESCRIPTION. IDDN.FR.001.270047.000.S.P.2003.000.21000.
- ASARES is a symbolic machine learning system that automatically infers, from descriptions of pairs of linguistic elements found in a corpus in which the components are linked by a given semantic relation, corpus-specific morpho-syntactic and semantic patterns that convey the target 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 is a audio/video frame server. IDDN.FR.001.320006.000.S.P.2006.000.40000,
- NAVITEX is a video annotation tool. IDDN.FR.001.190034.000.S.P.2007.000.40000,
- TELEMEX is a web service that enables TV and radio stream recording.
- VIDSIG computes a small and robust video signature (64 bits per image).
- VIDSEG computes segmentation features such as cuts, dissolves, silences in audio track, changes of ratio aspect, monochrome images. IDDN.FR.001.250009.000.S.P.2009.000.40000 ,
- ISEC, web application used as graphical interface for image searching engines based on retrieval by content.
- GPU-KMEANS, implementation of k-means algorithm on graphical process unit (graphic cards)
- CORRESPONDENCE ANALYSIS computes a factorial correspondence analysis (FCA) for image retrieval.
- GPU CORRESPONDENCE ANALYSIS is an implementation of the previous software Correspondence Analysis on graphical processing unit (graphical card).
- CAVIZ is an interactive graphical tool that allows to display and to extract knowledge from the results of a Correspondence Analysis on images.
- KIWI (standing for Keywords Extractor) is mostly dedicated to indexing and keyword extraction purposes.

- TOPIC SEGMENTER, is a software dedicated to topic segmentation of texts and (automatic) transcripts.
- S2E (Structuring Events Extractor) is a module which allows the automatic discovery of audiovisual structuring events in videos.
- 2PAC builds classes of words of similar meanings (“semantic classes“) specific to the use that is made of them in that given topic. IDDN.FR.001.470028.000.S.P.2006.000.40000
- FAESTOS (Fully Automatic Extraction of Sets of keywords for TOpic characterization and Spotting) is a tool composed of a sequence of statistical treatments that extracts from a morpho-syntactically tagged corpus sets of keywords that characterize the main topics that corpus deals with. IDDN.FR.001.470029.000.S.P.2006.000.40000
- FISHNET is an automatic web pages grabber associated with a specific theme.
- MATCH MAKER, semantic relation extraction by statistical methods.
- IRISAPHON produces phonetic words.
- PYTHON-GEOHASH is an implementation of the Geometric Hashing algorithm of [90] to check if geometrical consistency between pairs of images.
- AVSST is an Automatic Video Stream Structuring Tool. First, it allows the detection of repetitions in a TV stream. Second, a machine learning method allows the classification of programs and inter-programs such as advertisements, trailers, etc. Finally, the electronic program guide is synchronized with the right timestamps based on dynamic time warping. A graphical user interface is provided to manage the complete workflow.
- TVSEARCH is a content based retrieval search engine used to search and propagate manual annotation such as advertisement in a TV corpora.
- SAMUSA detects speech and/or musical segment in multimedia content.
- KERTRACK is a visual graphical interface for tracking visual targets based on particle filter tracking or based on mean-shift.
- MOZAIC2D creates of spatio-temporal mosaic based on dominant motion compensation.

5.2. Demonstration: Texmix

Participants: Morgan Bréhinier, Sébastien Campion [correspondent], Guillaume Gravier.

The gradual migration of television from broadcast diffusion to Internet diffusion offers tremendous possibilities for the generation of rich navigable contents. However, it also raises numerous scientific issues regarding de-linearization of TV streams and content enrichment. In this Texmix demonstration, we illustrate how speech in TV news shows can be exploited for de-linearization of the TV stream. In this context, de-linearization consists in automatically converting a collection of video files extracted from the TV stream into a navigable portal on the Internet where users can directly access specific stories or follow their evolution in an intuitive manner.

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.

The demonstration was presented in several workshops (Futur en Seine - Paris, Futur TV - Berlin, ICMR - Hong Kong, French Minister for higher education and research - Rennes, RFIA - Lyon) and a video has been made available online on the portal of the EIT ICT Labs OpenSEM project. An article about this demonstrator was also published in ’Emergences’ http://emergences.inria.fr/2012/newsletter-n22/L22-TEXMIX.

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 our data and to carry out our experiments. In 2005, we began some 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).

In 2008, we build up a corpus of multimedia data. It consists in a continuous recording (6 months) of two TV channels and three radios. It also includes web pages related to these contents captured on broadcaster's website. This corpus is to be used for different studies like the treatment of news along the time and to provide sub-corpus like TV news within the Quaero project (see below). The manual annotation of all the TV programs is under progress. 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, in particular to improve the speed of building index or language model. The previous server dedicated to this kind of jobs (acquired in 2008) has been upgraded to 96GB of RAM.

This year, 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.

This platform is funded by a joint effort of Inria, INSA Rennes and University of Rennes 1.

WILLOW Project-Team

5. Software

5.1. Patch-based Multi-view Stereo Software (PMVS)

PMVS is a multi-view stereo software that takes a set of images and camera parameters, then reconstructs 3D structure of an object or a scene visible in the images. Only rigid structure is reconstructed. The software outputs a set of oriented points instead of a polygonal (or a mesh) model, where both the 3D coordinate and the surface normal are estimated at each oriented point. The software and its documentation are available at http://www.di.ens.fr/pmvs/. The software is distributed under GPL. A US patent corresponding to this software "Match, Expand, and Filter Technique for Multi-View Stereopsis" was issued on December 11, 2012 and assigned patent number 8,331,615.

5.2. SPArse Modeling Software (SPAMS)

SPAMS v2.3 was released as open-source software in May 2012 (v1.0 was released in September 2009 and 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/.

5.3. 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 6.4.5 . 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.

5.4. Automatic Mining of Visual Architectural Elements

The code on automatic mining of visual architectural elements (v4.3) described in (Doersch *et al.* SIGGRAPH [6]) has been publicly released online in December 2012 (earlier version v3.0 was released in September 2012) at http://graphics.cs.cmu.edu/projects/whatMakesParis/paris_sigg_release.tar.gz.

5.5. Automatic Alignment of Paintings

The code for automatic alignment of paintings to a 3D model (Russell et al. 2011) was made publicly available in October 2012 at http://www.di.ens.fr/willow/research/paintingalignment/index.html.

5.6. Multi-Class Image Cosegmentation

This is a package of Matlab code implementing multi-class cosegmentation (Joulin *et al.* CVPR 2012 [13] and unsupervised discriminative clustering for image co-segmenting (Joulin *et al.* CVPR 2010) and (Joulin *et al.* NIPS 2010). The aim is to segment a given set of images containing objects from the same category, simultanously and without prior information. The package was last updated in September 2012 and is available at http://www.di.ens.fr/~joulin/code/DALCIM.zip.

5.7. Convex Relaxation of Weakly Supervised Models

This is a package of Matlab code implementing a general multi-class approach to weakly supervised classification described in (Joulin and Bach ICML 2012 [12]). The goal is to avoid local minima typically occuring expectation-maximization like algorithms and to optimize a cost function based on a convex relaxation of the soft-max loss. The package was last updated in September 2012 and is available at http://www.di.ens.fr/~joulin/code/ICML12_Joulin.zip.

5.8. Non-uniform Deblurring for Shaken Images

An online demo of non-uniform deblurring for shaken images implementing the algorithm described in [8] and (Whyte et al. CPCV 2011) was made available in 2012 at http://www.di.ens.fr/willow/research/deblurring/. The demo takes as an input an image uploaded by the user, automatically estimates the blur, and outputs the deblurred image.