A platform dedicated to knowledge engineering for the development of image processing applications

Arnaud Renouf, Régis Clouard, Marinette Revenu

A PLATFORM DEDICATED TO KNOWLEDGE ENGINEERING FOR THE DEVELOPMENT OF IMAGE PROCESSING APPLICATIONS

Arnaud Renouf, Régis Clouard, Marinette Revenu
GREYC Laboratory, CNRS UMR 6072
6, Boulevard Maréchal Juin, 14050 CAEN Cedex, France
{Arnaud.Renouf, Régis.Clouard, Marinette.Revenu}@greyc.ensicaen.fr

Keywords: Image processing, application formulation, knowledge engineering, ontology

Abstract: In this paper, we propose a platform dedicated to the knowledge extraction and management for image processing applications. The aim of this platform is a knowledge-based system that generates automatically applications from problem formulations given by inexperienced users. We also present a new model for the formulation of such applications and show its contribution to the platform performance.

1 INTRODUCTION

In the last fifty years, a lot of image processing applications have been developed in many fields (medicine, geography, robotic, industrial vision, ...). We know that image processing specialists design applications by trial errors cycles. They do not enough reuse already developed solutions and design new ones nearly from scratch. The lack of application formulation modeling and formalization is a reason of this behavior. Indeed, image processing experts do not realize a complete and rigorous formulation of the applications. Only the solutions are used as their definitions. Therefore, the reusability of these applications is very poor because the limits of the solution applicability are not explicit. Moreover they often suffer from a lack of modularity and the parameters are also often tuned manually without giving explanations on the way they are set.

Knowledge based systems such as OCAPI (Clément and Thonnat, 1993), MVP (Chien and Mortensen, 1996) or BORG (Clouard et al., 1999) were developed to construct automatically image processing applications and to make explicit the knowledge used to solve such applications. However, a priori knowledge on the application context (sensor effects, noise type, lighting conditions, ...) and on the goal to achieve was more or less implicitly encoded in the knowledge base. This implicit knowledge restricts the range of application domains for these systems and it is one of the reasons of their failure (Draper et al., 1996).

More recent approaches bring more explicit modelling (Nouvel and Dalle, 2002) (Maillot et al., 2004) (Hudelot and Thonnat, 2003) (Bombardier et al., 2004) (Town, 2006) but they are all limited to the description of business objects for detection, segmentation, image retrieval, image annotation or recognition purposes. Some of them use ontologies that provide the concepts needed for this description: a visual concept ontology for object recognition in (Maillot et al., 2004), a visual descriptor ontology for semantic annotation of images and videos in (Bloehdorn et al., 2005) or image processing primitives in (Nouvel and Dalle, 2002) (Hudelot and Thonnat, 2003). Others capture the business knowledge through meetings with the specialists: use of the NIAM/ORM method in (Bombardier et al., 2004) to collect and map the business knowledge to the vision knowledge. But they do not completely tackle the problem of the application context description (or briefly as in (Maillot et al., 2004)) and the effect of this context on the images (environment, lighting, sensor, image format). Moreover they do not define the means to describe the image content when objects are a priori unknown or unusable (e.g. in robotic, image retrieval or restoration applications). They also suppose that the objectives are well known (to detect, to extract or to recognize an object with a restrictive set of constraints) and therefore they do not address their specification.

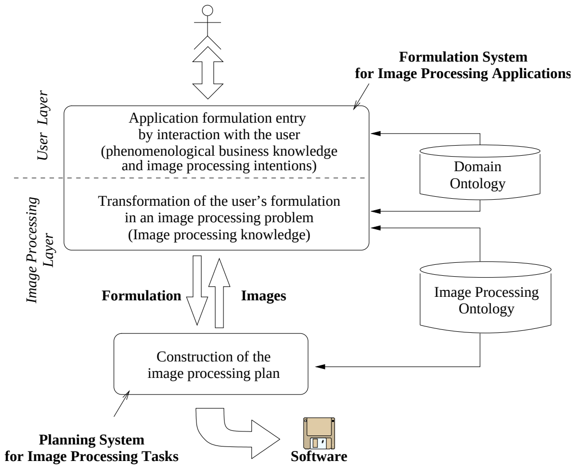
Formulation System
for Image Processing Applications
User Layer
Application formulation entry
by interaction with the user
(phenomenological business knowledge
and image processing intentions)
Domain
Ontology
Image Processing
Layer
Transformation of the user's formulation
in an image processing problem
(Image processing knowledge)
Formulation
Images
Image Processing
Ontology
Construction of the
image processing plan
Planning System
for Image Processing Tasks
Software

Figure 1: The system architecture of the project.

To overcome these problems, we aim at building a methodology and a guideline for the development of such applications in order to make it easier and more reliable. To reach this goal, we have to make explicit the formulation of the problem to be solved, and the knowledge used by image processing experts during the design. In this paper, we propose a complete platform dedicated to these objectives. It includes a system that generates automatically image processing applications from formulations given by inexperienced users. First of all, we introduce the global project (Section 2) and then we present the various platform components and the actors (Section 3). Section 4 defines briefly our formulation model for image processing applications and Section 5 its contribution to the platform performance. Finally, we conclude on the contribution of this work for the image processing field.

2 THE PANTHEON PROJECT

Our work is part of the Pantheon project which aims at developing a system that automatically generates image processing softwares from user-defined problem formulations. This system is composed of two sub-systems (Fig. 1): a formulation system for image processing applications which is the focus of our work, and a planning system for image processing tasks (Clouard et al., 1999). The user defines the problem with the terms of his/her domain by interaction with the user layer of the formulation system. This part of the system is a human-machine interface which uses a *domain ontology* to handle the information dedicated to the user. It groups concepts that allow the users to formulate their processing intentions and define the image class. Then the formulation system translates this user formulation into image processing terms taken from an *image processing ontology*. This translation achieves the mapping between the phenomenological domain knowledge of the user and the image processing knowledge.

The result of this translation is an image processing request which is sent to the planning system to generate the program that responds to this request. This cooperation needs the two sub-systems to share the *image processing ontology*. Then the formulation system executes the generated program on test images and presents the results to the user for evaluation purposes. The user is responsible for modifying the request. This process is repeated until the results are validated.

In the next section, we present the platform created to help image processing specialists in the design of their applications and in the formalization of the knowledge involved in this activity.

3 PLATFORM ARCHITECTURE

The platform is composed of two co-dependent parts (Figure 2):

- the left part is the knowledge-based system itself. It generates automatically an image processing software that satisfies specific user's requirements.
- the right part is the programming environment devoted to image processing experts. This environment provides a methodological guide and programming facilities to make the development easier and more reliable.

The key idea of this distinction is to reuse the results capitalized during the programming process to reinforce the knowledge-based system, and conversely to experiment the tools and the methodology within the knowledge base to reinforce the acquisition methodology.

3.1 Programming Environment Part

The programming environment is composed of three components (Figure 2):

- **Pandore** is a library of image processing operators and a programming environment which allows to construct applications by writing scripts that supervise the execution of the operators.
- **Ariane** is a visual programming interface that provides an ergonomic way to develop applications. It makes easier the composition of the

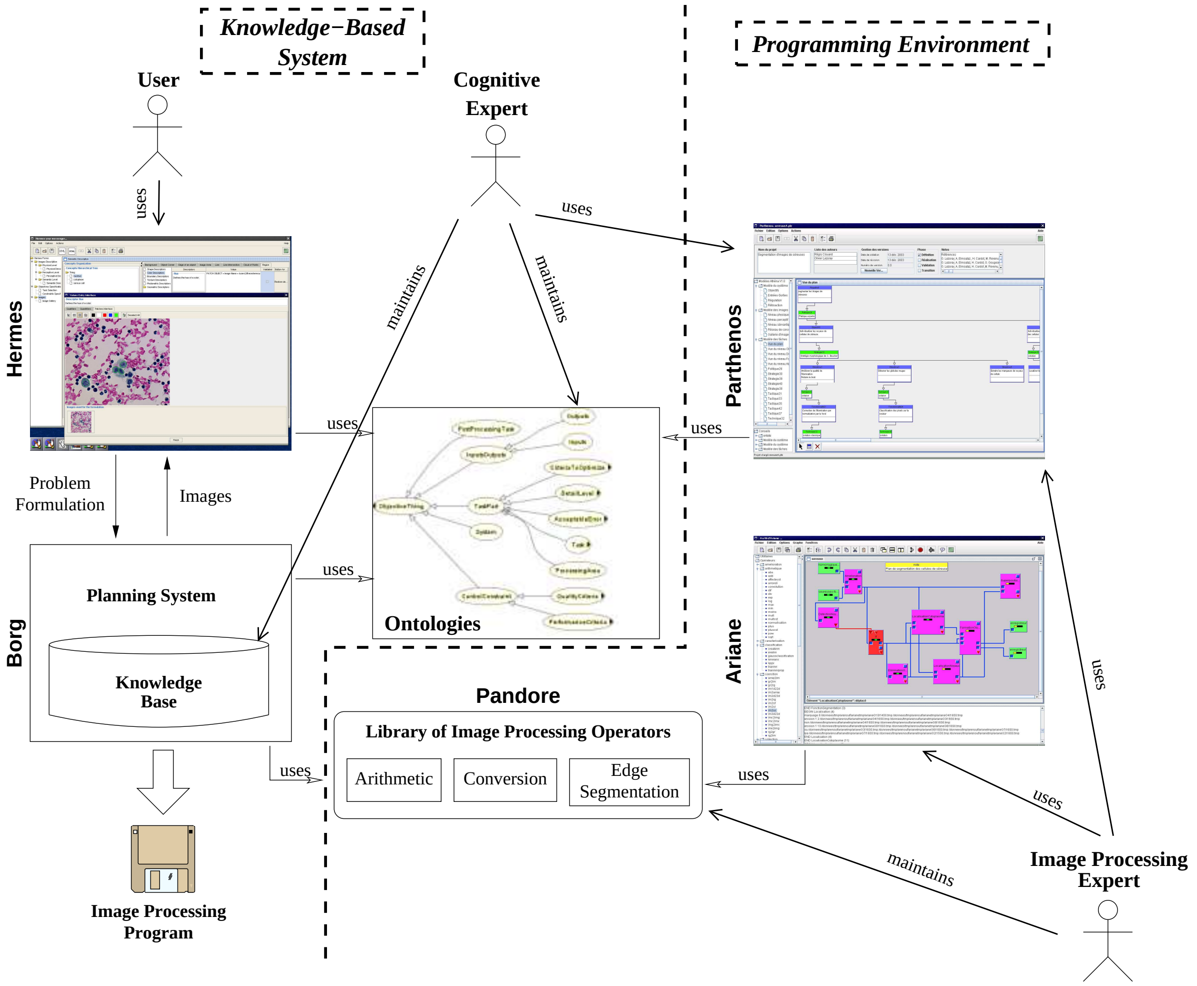

Figure 2: The platform architecture

different operators thanks to an intuitive interface that proposes to design the application using graphical objects (Figure 2).

- **Parthenos** is a CASE tool for the development of image processing programs. It helps the image processing experts to the rationalization of the different steps of the design process by asking him/her to justify the choices (Figure 2).

3.2 Knowledge-based System Part

The knowledge-based system part is composed of two main components and ontologies:

- **Borg** (Clouard et al., 1999) is a knowledge based system that generates suitable image processing programs thanks to competencies encoded in its knowledge base.

- **Hermes** is a user interface that allows end-users' specifications of image processing applications. It leads the user to give a formalized formulation through a human-machine interface (Figure 2).

3.3 Actors

This platform takes under consideration three categories of actors:

- **the image processing expert** develops new applications and models existing ones. S/he has to explain every choices made during the design using the case tool Parthenos. S/he can add new operators to the library if needed.

- **the cognitive expert** uses the results of the modeling obtained with Parthenos to extract the knowledge involved in the construction of solutions and, finally, maintains the ontologies and the knowledge base.

- **the end user** is inexperienced in the image processing field but s/he is an expert of the application domain. S/he specifies objectives, describes the effects of the acquisition system on the resulting images and the images content in a relevant way. S/he is also in charge of the evaluation of the results to validate the solution.

4 FORMULATION MODEL

We notice that the formulation of image processing applications has been little studied. However a complete and rigorous formulation of these applications is essential towards the goal of designing more robust and more reliable vision systems, and in order to fix the limits of the application, to favor reusability and to enhance the evaluation. Such a formulation has to clearly specify the objective and to identify the range of the considered input data. Unfortunately, formulating an image processing application is a problem of qualitative nature that relies on subjective choices. Hence, an exhaustive or exact definition of the problem does not exist. Only an approximative characterization of the desired application behavior can be defined.

Our approach tends to capture the phenomenological business knowledge from a user and to map this knowledge to image processing knowledge used to find a solution. From this consideration, we studied in a first step the formulation from an image processing expert point of view to create a model (and its formalization through an ontology) for image processing applications. Then we looked for means to capture the user's knowledge (the domain knowledge) and to translate it into information useful for the planning system. A description of this part can be found in (Renouf et al., 2007).

The proposed formulation model identifies and organizes the relevant information which are necessary and sufficient for the planning system or an image processing specialist to develop a convenient solution. It covers all image processing tasks and is independent of any application domain since the business knowledge is acquired from the user. It is composed of two parts: the objectives specification and the image class definition. We present here a brief review of this model formalization through an *image processing ontology* (notice that this ontology tackles the problem of the formulation and does not represent the whole image processing field).

4.1 Objectives Specification

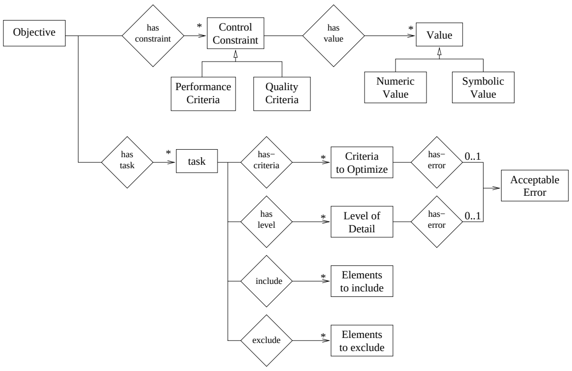

Figure 3: CML representation of the objective specification model.

The objectives specification relies on the teleological hypothesis. It leads us to define an image processing application through its finalities. Hence, the objectives are specified by a list of control constraints and a list of tasks with related constraints (criteria to optimize with acceptable errors, levels of detail with acceptable errors, elements to be excluded and included in the result). We present on Fig. 3 the formalization of the model for the objectives specification using the CML[1] formalism of CommonKADS (Schreiber et al., 1994). It represents the highest level of the model and does not show every concept of this ontology part. A concept like *Task* is the root of a hierarchical tree where nodes are processing categories and leaves are effective tasks. Processing categories belong to reconstruction, segmentation, detection, restoration, enhancement and compression objectives. Effective tasks are 'Extract object', 'Detect object', 'Enhance', 'Correct', *etc*. Some of these tasks have to be associated with an element of the image class definition: *e.g.* 'Enhance' with an instance of the sub-concepts of 'Acquisition Effect', 'Extract object' with an instance of a 'Business Object'.

By creating instances and setting their properties in the *image processing ontology* (it can be done through Parthenos or Hermes), image processing experts can formalize the formulation of image processing application objectives. Hermes also instantiates this ontology according to the choices of the user in the interface. An example of the formulation obtained on a cytological application where the objective is to extract serous cell nuclei can be found in (Renouf et al., 2007).

4.2 Image Class Definition

The image class definition relies on two hypotheses: the semiotic and the phenomenological ones. The semiotic hypothesis leads us to define the image class considering three levels of description:

- the physical level focuses on the characterization of the acquisition system effects on the images.

[1] Conceptual Modeling Language

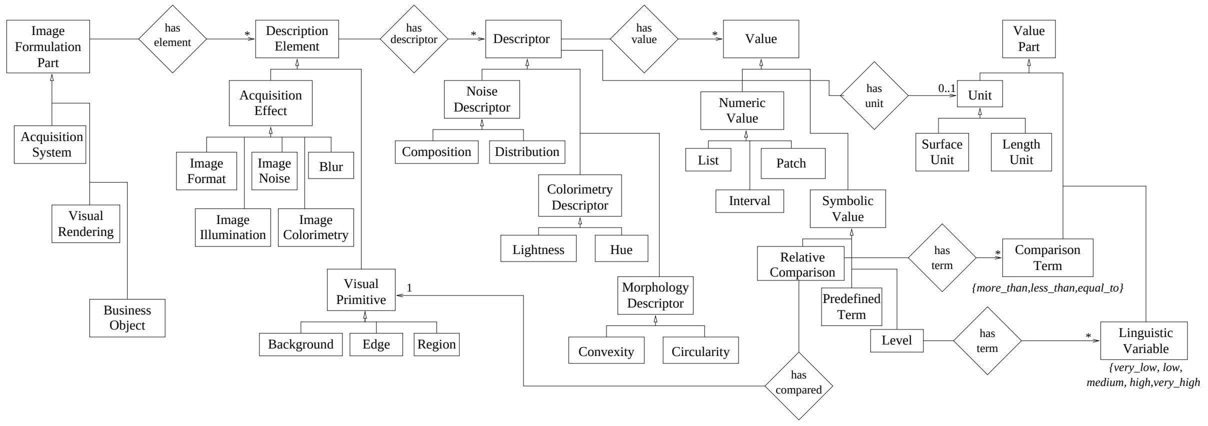

Figure 4: CML representation of the image class definition model.

- the perceptive level focuses on the description of the visual primitives (regions, lines, background, ...) without any reference to the business objects.
- the semantic level focuses on the identification of the business objects visualized in the images.

The phenomenological hypothesis states that a visual characterization of the images and the business objects are sufficient to design image processing applications. Hence, we do not need a complete description of the objects given by the user but only a visual description. The system asks the user to describe how the objects appear in the images but not what they are. For example, in the case of aerial images, cars are only defined as homogeneous rectangular regions.

We present on Figure 4 the formalization of the image class definition model using the CML representation. Here again, this representation is only a part of the ontology and does not show all the descriptors (color, texture, shape, geometric, photometric descriptors), visual primitives (region, edge, background, point of interest, points cloud), types of values (symbolic and numeric values) and spatial and composition relations. In the same way as for the objectives specification, image processing experts and Hermes use this part of the ontology to define the image class (See (Renouf et al., 2007) for an example).

5 CONTRIBUTION

Prior to the design of the formulation formalization through ontologies, the image processing experts working on the platform had to justify their choices within Parthenos using natural language. These information are useful to understand the reasoning during the design process but the cognitive expert has then to study and formalize these information to feed the knowledge base of the planning system. Moreover, to use this planning system, the formulation of a new application had to be made by the image processing expert thanks to meetings with the user.

The *image processing ontology* defines the necessary and sufficient information to allow the formulation of applications from the image processing expert point of view. The image processing experts involved in the project use this ontology for their tasks of engineering of new applications and re-engineering of existing ones. They build the formulation and have to justify the choices made during the design process by using elements of this formalized formulation (using criteria pro and con at each step of the decomposition of the image processing plan). Thus, they can verify that the information justifying the choices are included in the formulation. Such a work of re-engineering is very interesting because it makes explicit the knowledge used tacitly by the application designers. Besides, it often reveals weaknesses on the limits of applicability of the considered applications.

These experiments also permit to put the formulation model to the proof and discover the missing concepts of the ontologies which can be added by the cognitive expert. Moreover, the cognitive expert can easily feed the knowledge base because s/he has the conditions of applicability of the different application parts. Actually, it enhances the collaborative work of these two kinds of experts because they now use the same language which is fixed by the ontology.

The *domain ontology* (used by Hermes) gives the concepts used to formulate the applications from the application domain expert point of view. Hermes uses the restrictions defined on the properties of the ontology. This one is implemented in OWL DL and restrictions are expressed using description logics: for example, we defined a restriction on the property 'hasDescriptor' of the concept 'Background' that limits

its range of values only to ('TextureDescriptor', 'ColorDescriptor' or 'PhotometricDescriptor'). During the user's formulation, the interface is built dynamically according to the user's choices and the ontology content. Therefore, the interface is updated as soon as new concepts are introduced in the ontology by the cognitive expert. The formulation system also uses inference rules to propose default values to the user. For example, at the physical level, it proposes types of noise and defects that often degrade images according to the type of acquisition system.

We conduct experiments with inexperienced users in the image processing field. They are asked to formulate a problem defining an application using the human-machine interface. This work allows us to check if the concepts of the *domain ontology* are handleable for this kind of users and to enhance the interface ergonomics. It also reveals the difficulties encountered during the act of formulation.

Some recent works use application ontologies to represent visual properties in order to solve a problem of vision (*e.g.* in (Koenderink et al., 2006) to assess the quality of young tomato plant, in (Bombardier et al., 2004) to classify wood defects). These ontologies are built through meetings between a domain expert and an application designer, and they are specific to the task to be performed. Such ontologies can be constructed using our system and used, at least, by the image processing part of the considered application.

6 CONCLUSION

Our platform allows to study the image processing knowledge used in the development of applications. It is complete since it allows to formulate the problems, to model the solutions and to rationalize the design process during their development. Its different components help the actors of the platform in their work and the ontologies permit an effective collaborative work through their central role.

This work is a contribution to the image processing field because the modeling of the formulation allows to give prominence to the knowledge used in the development of such applications. It defines a guideline on the way we have to tackle such applications and identifies their formulation elements. The explicitness of these elements is very useful to acquire the image processing knowledge used by the planning system: they are used to justify the choice of the algorithms regarding the application context and therefore to define the conditions of applicability of the image processing techniques. Hence, it also enhances the evaluation and favors the reusability of solution parts.

REFERENCES

Bloehdorn, S., Petridis, K., Saathoff, C., Simou, N., Tzouvaras, V., Avrithis, Y., Handschuh, S., Kompatsiaris, Y., Staab, S., and Strintzis, M. G. (2005). Semantic annotation of images and videos for multimedia analysis. In *ESWC*, volume 3532 of *LNCS*, pages 592–607. Springer.

Bombardier, V., Lhoste, P., and Mazaud, C. (2004). Modélisation et intégration de connaissances métier pour l'identification de défauts par règles linguistiques floues. *Traitement du Signal*, 21(3):227–247.

Chien, S. and Mortensen, H. (1996). Automating image processing for scientific data analysis of a large image database. *IEEE PAMI*, 18(8):854–859.

Clément, V. and Thonnat, M. (1993). A knowledge-based approach to integration of image procedures processing. *CVGIP: Image Understanding*, 57(2):166–184.

Clouard, R., Elmoataz, A., Porquet, C., and Revenu, M. (1999). Borg : A knowledge-based system for automatic generation of image processing programs. *IEEE PAMI*, 21(2):128–144.

Draper, B., Hanson, A., and Riseman, E. (1996). Knowledge-directed vision : Control, learning, and integration. In *Proc. of IEEE*, volume 84, pages 1625–1681.

Hudelot, C. and Thonnat, M. (2003). A cognitive vision platform for automatic recognition of natural complex objects. In *Proc. of the 15th IEEE ICTAI*, page 398, Washington, DC, USA. IEEE Computer Society.

Koenderink, N. J. J. P., Top, J. L., and van Vliet, L. J. (2006). Supporting knowledge-intensive inspection tasks with application ontologies. *Int. J. Hum.-Comput. Stud.*, 64(10):974–983.

Maillot, N., Thonnat, M., and Boucher, A. (2004). Towards Ontology Based Cognitive Vision (Long Version). *Machine Vision and Applications*, 16(1):33–40.

Nouvel, A. and Dalle, P. (2002). An interactive approach for image ontology definition. In *13ème Congrès de Reconnaissance des Formes et Intelligence Artificielle*, pages 1023–1031, Angers, France.

Renouf, A., Clouard, R., and Revenu, M. (2007). How to formulate image processing applications ? In *Proceedings of the International Conference on Computer Vision Systems*, Bielefeld, Germany.

Schreiber, G., Wielinga, B., Akkermans, H., Van de Velde, W., and Anjewierden, A. (1994). CML: The CommonKADS Conceptual Modelling Language. In Steels, L., Schreiber, G., and de Velde, W. V., editors, *EKAW 94*, volume 867 of *Lecture Notes in Computer Science*, pages 1–25, Hoegaarden, Belgium. Springer Verlag.

Town, C. (2006). Ontological inference for image and video analysis. *Mach. Vision Appl.*, 17(2):94–115.