Ontology Based Object Learning and Recognition

Nicolas Maillot

UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS   UFR Sciences

École Doctorale STIC

Thèse

pour obtenir le titre de
Docteur en Sciences
de l'Université de Nice - Sophia Antipolis

Spécialité : INFORMATIQUE

présentée et soutenue par

Nicolas MAILLOT

Ontology Based Object Learning and Recognition

Thèse dirigée par Monique THONNAT
Équipe d'accueil : ORION – INRIA Sophia-Antipolis

Soutenue publiquement le 14 Décembre 2005
devant le jury composé de :

Mme	Annie	CAVARERO	Pr, Université de Nice	Président
Mme	Nozha	BOUJEMAA	DR, INRIA Rocquencourt, France	Rapporteur
M.	Anthony	COHN	Pr, University of Leeds, UK	Rapporteur
Mr	Philippe	MULHEM	CR, CNRS Grenoble, France	Examinateur
Mr	Phil	CULVERHOUSE	Dr, University of Plymouth, UK	Examinateur
Mme	Monique	THONNAT	DR, INRIA Sophia Antipolis, France	Directrice

Acknowledgements

I would like to thank Pr Anthony Cohn for accepting to review this manuscript. Many thanks as well to Dr Phil Culverhouse for being an examiner in the thesis comittee. It was a real pleasure to help him writing the EVA proposal. I will now switch to French.

Merci à Nozha Boujemaa pour avoir accepté, malgré un emploi du temps très chargé, d'être rapporteur de ce manuscrit. J'adresse également mes remerciements à Philippe Mulhem pour sa participation à ce jury. Je tiens aussi à remercier Annie Cavarero pour avoir accepté de présider ce jury.

Merci à Monique Thonnat pour m'avoir suivi et soutenu tout au long de ces trois années de thèse. Elle m'a beaucoup transmis: des connaissances scientifiques évidemment, mais aussi une partie de sa tenacité, de son exigence, et de son optimisme. Tout cela me sera très utile à l'avenir.

Merci à Catherine Martin pour son aide précieuse sur tous les problèmes administratifs.

Merci à Alain Boucher de m'avoir si bien guidé durant le stage de DEA.

Je tiens à adresser une pensée toute particulière à Sabine Moisan qui m'a aidé au niveau des réalisations logicielles. J'espère la revoir le plus tôt possible parmi nous.

Merci à Jean-Paul Rigault de m'avoir confié une charge d'enseignement à l'ESSI. J'ai beaucoup appris à son contact.

Merci à Bernard Dubuisson, Professeur à l'Université de Technologie de Compiègne, qui est probablement le meilleur enseignant que je n'ai jamais eu. Son cours de reconnaissance des formes (SY19) a été pour moi une révélation.

Merci à Christophe Tornieri pour avoir tant échangé durant ces années passées dans le même bureau. Je lui souhaite bonne chance dans ses nouvelles aventures.

Merci à Bernard, Lan, et Nicolas Chleq pour leur relecture constructive de cette thèse.

Merci à tous les autres membres de l'équipe ORION pour leur bonne humeur et pour tous les bons moments et discussions (scientifiques ou non): Céline, Francois, Benoît, Nicolas Dey, Alberto, Frédéric, Magali, Julien, Vincent, Bernard, Florent, Valéry, Florence, Gabrielle, Marcos, et Annie.

Merci aussi à Adeline, Donavan, Marie-Julie, Jérôme, Géraldine, Franck, Nathalie, Thomas, et Élodie pour tous ces bons moments passés en dehors du travail.

Merci à mon ami d'enfance, Renaud. Ca fait si longtemps que je partage avec lui cette passion pour l'informatique. C'est lui qui m'a fait découvrir l'Internet puis Linux il y a plus de dix ans. Grâce à lui, j'ai rapidement su ce que je voulais faire dans la vie.

Je tiens aussi à remercier mes parents pour m'avoir toujours encouragé, soutenu, et poussé à réaliser mes rêves. Sans eux, rien de tout cela n'aurait été possible.

Enfin, merci à mon épouse, Jeanne-Marie, pour sa patience, son amour, et son soutien sans faille durant (et en dehors de) la période de rédaction. Je souhaite arriver un jour à lui rendre tout ce qu'elle m'a donné.

Abstract

This thesis deals with the problem of complex object recognition. The proposed approach takes place in the conceptual framework of cognitive vision. This thesis shows how an object categorization system is set up in three phases.

The knowledge acquisition phase consists of acquiring domain knowledge as a taxonomy/partonomy of domain classes. It also consists of acquiring the visual description of these domain classes. This description is driven by a visual concept ontology composed of several types of concepts (spatial concepts and relations, color concepts and texture concepts). Each visual concept of the ontology is associated with low-level features and algorithms. The visual concept ontology stands as a user-friendly interface between expert knowledge and image processing level.

The learning phase results in a set of visual concept detectors. The role of a visual concept detector is to detect visual concepts used during knowledge acquisition in any image. A visual concept detector is obtained by training Support Vectors Machines with features extracted in segmented image samples labeled by visual concepts.

The categorization phase uses both the acquired domain knowledge and the visual concept detectors obtained during the learning phase. Domain knowledge is used to generate hypotheses which have to be verified in the image by visual concept detection in automatically segmented images. The categorization result is composed of the objects recognized in the image with their visual description.

The approach has been applied to the problem of semantic image indexing and retrieval.

keywords: Ontology, Knowledge Acquisition, Object Recognition, Machine Learning, Image Indexing, Image Retrieval.

Résumé

Cette thèse se place dans le cadre du problème de la reconnaissance d'objets et plus généralement dans celui de la vision cognitive. L'approche proposée se décompose en trois phases principales:

Une phase d'acquisition de connaissances qui consiste à acquérir la connaissance d'un domaine d'application sous la forme d'une hiérarchie de classes d'objets et de sous parties. Il s'agit également de décrire ces classes du domaine en termes de concepts visuels (forme, texture, couleur, relations spatiales) fournis par une ontologie. Chaque concept visuel de cette ontologie étant associé à des descripteurs bas niveau, le fossé sémantique est réduit de manière conviviale pour un expert.

La phase d'apprentissage consiste, à partir d'images d'exemples segmentées et labellisées, à obtenir un ensemble de détecteurs de concepts visuels. Ces détecteurs sont obtenus par l'entrainement de Support Vector Machines avec les descripteurs numériques extraits dans les images d'exemples segmentées et labellisées par des concepts visuels.

La phase de catégorisation utilise la connaissance acquise ainsi que les détecteurs de concepts visuels obtenus lors de la phase d'apprentissage. La connaissance sert à générer des hypothèses qui doivent être vérifiées dans l'image à interpréter. Cette vérification consiste à détecter des concepts visuels dans l'image segmentée automatiquement. Le résultat de la catégorisation est exprimé en termes de classes du domaine mais aussi en termes de concepts visuels.

L'approche proposée a notamment été utilisée été utilisée pour l'indexation et la recherche sémantique d'images.

Mots-Clés: Ontologie, Acquisition de Connaissances, Reconnaissance d'Objets, Apprentissage Artificiel, Indexation d'Images, Recherche d'Images.

Contents

Chapter 1

Introduction

Object recognition has been an heavily studied research topic for about three decades. It is an extremely challenging problem. Despite some success in restricted domains (e.g. quality control, face recognition), automatic object recognition remains an open problem. Most systems developed so far suffer from a lack of flexibility and adaptability. This statement is also true for contemporary computer vision in general.

In order to improve the situation, a new discipline called *cognitive vision* has been introduced. A research roadmap of cognitive vision can be found in [Auer et al., 2005]. As explained in this roadmap, *the term cognitive vision has been introduced to encapsulate an attempt to achieve more robust, resilient, and adaptable computer vision systems by endowing them with a cognitive faculty: the ability to learn, adapt, weigh alternative solutions, and even the ability to develop new strategies for analysis and interpretation.*

Our focus is the object recognition problem. We tackle this problem based on the following scientific foundations of cognitive vision: *recognition*, *architecture*, *representation*, *learning* and *communication*. This manuscript shows how all these foundations can be used as a basis for building an operational object recognition system.

Advances in object recognition have applications in the following fields:

- **Image and video indexing and retrieval**. The huge amount of multimedia content produced by personal cameras and video recorders or available on the World Wide Web has created an urgent need for efficient techniques for accessing this content.

- **Video Monitoring and Surveillance**. Video monitoring and surveillance often imply recognizing scenarios (e.g. bank attack, fighting) involving complex objects (e.g. human, car, luggage, aircraft). Most scenario recognition algorithms (e.g. [Vu et al., 2003]) make the assumption that objects involved in the scene are correctly recognized. Thus, reliable object recognition is necessary to achieve efficient and reliable monitoring and surveillance.

- **Robotics**. Scene interpretation and object recognition capabilities are very important for robotic systems. In [Auer et al., 2005], it is explained that a robotic mobile agent should be able to achieve acquisition of the model of the physical environment, object identification and understanding of their function but also detection of new objects. Sophisticated object identification and recognition capabilities are necessary to achieve these tasks.

- **Smart environments**. In [Campbell and Krumm, 2000], the importance of object recognition in the context of smart rooms is stressed. While staying at home or at their office, people interact together but also with many objects (e.g. keyboard, remote control, phone, book). An example of functionality of a smart room is the functionality of finding (lost) objects in a room (e.g. remote control, keys). Another important functionality of smart rooms is their ability to infer intents and actions of the user (e.g. turning on the lights when a user starts reading a book). In both cases, efficient object recognition mechanisms are required.

1.1 Context of the study

This work has been conducted in the Orion team located at Inria Sophia Antipolis. Orion is a multi-disciplinary team at the frontier of computer vision, artificial intelligence and software engineering. The team has accumulated a strong expertise in these areas throughout the years.

One topic of particular interest for the team is knowledge-based image and video understanding. A representative work can be found in [Thonnat and Bijaoui, 1989]: in this approach, a knowledge-based approach is used for galaxy recognition purposes. In [Vu et al., 2003], a new algorithm for real-time recognition of temporal scenario is presented.

Some contributions have also been made by the team in the software engineering community. In particular, *program supervision* has been introduced for improving the management of image processing libraries [Shekhar et al., 1998]. In [Ossola et al., 1996], Ossola proposes a software platform that combines two knowledge-based systems in cooperative way. The first one is dedicated to high-level image interpretation and the second one to supervision of image processing programs. More recently, in [Hudelot, 2005], Hudelot proposes a cognitive vision platform that contains a knowledge-based system dedicated to image data management. This system works in cooperation with two other knowledge-based systems respectively dedicated to high-level interpretation and to supervision of image processing programs.

1.2 Problem Statement and Objectives

Our goal is semantic image interpretation for complex object classification purposes. The difficulty of semantic image interpretation can be illustrated by fig. 1.1. Indeed, this image can be interpreted as a light object on a dark background. One can also see an astronomical object and more precisely a spiral galaxy. This illustrates that semantic interpretation relies on a priori knowledge. This means that semantics is not inside the image but results from the association of perception with a priori knowledge acquired through experience. Depending on his/her experience, this association can be made differently from a person to another. This is also called the *polysemy* of the image. As explained in [Shatford, 1986], "the delight and frustration of pictorial resources is that a picture can mean different things to different people". In [Barthes, 1977], Barthes differentiates denoted and connoted meanings. The denoted meanings of an image, or denotation, are equivalent to our perceived reality. Barthes calls this the analogon. The connotative meanings, which are at least partly constructed by the treatment of the image, are the sum of responses to the meanings of the image. All these responses are historically derived and culturally specific. A very interesting study on the influence of image polysemy in video retrieval is proposed in [Christel and Hauptmann, 2005].

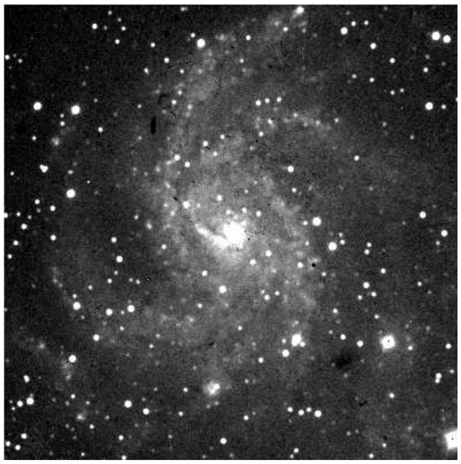


Figure 1.1: The semantic interpretation of this image requires a priori knowledge in astronomy

Figure 1.2 is particularly interesting and shows the importance of the context. It was used as an advertising campaign by the french newspaper *Le Monde* which insists on the fact that the meaning of an image is outside of the image (i.e. in this case, in the text authored by journalists). It shows three images of the same event. The top image can only be interpreted as an handshake which is a code and a rule of politeness. Is is hard to say more about it. In the middle image, a little bit more can be said: It seems to be an agreement between three persons. One of them wears a scarf which gives an international connotation to the image. One might ask if this image represents agreement related to the oil business. The image in the bottom has a completely different meaning, the presence of Bill Clinton combined with this handshake changes the interpretation of this image: by being aware of this political context, the image can then be interpreted as the result of a negotiation related to critical political issues. This interpretation requires a priori

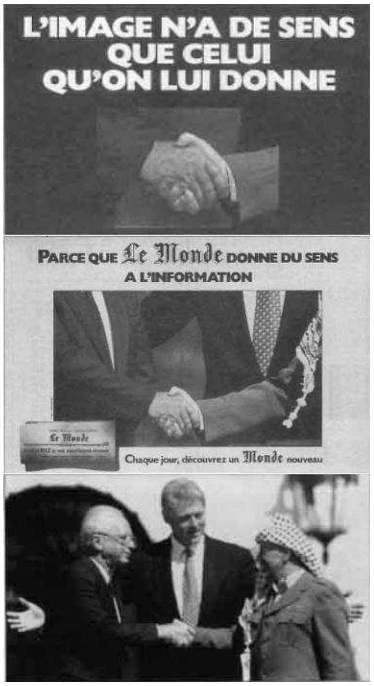

L'IMAGE N'A DE SENS
QUE CELUI
QU'ON LUI DONNE
PARCE QUE Le Monde DONNE DU SENS
A L'INFORMATION
Le Monde
Chaque jour, découvrez un Monde nouveau

Figure 1.2: Three images of the same event: the Oslo accords (1993). Recognizing the people in the bottom image creates a political context that gives a particular meaning to this image.

knowledge about international politics. Note that this image could be given to a child who is a few years and he/she would probably not be able to tell more than a knowledgeable adult on the middle image.

These observations lead to several questions:

- How to acquire a priori knowledge required for enabling useful (i.e. answering the needs of the end-users in a given context) recognition tasks? This question is directly related to knowledge acquisition issues and especially to the well-known knowledge acquisition bottleneck.

- How to reproduce the experimental process that grounds knowledge with perception ? One important underlying issue is to reduce the semantic gap between high-level knowledge and low-level image features and algorithms.

- How to use the grounded knowledge for enabling efficient recognition? In other words, once knowledge has been acquired and grounded, how can it be used as a support for object recognition.

Our work aims at solving the problems listed above by the following contributions:

- We propose to use the advances made in the knowledge engineering community to reduce the knowledge acquisition bottleneck. In particular a visual concept ontology is introduced and enables the reduction of the gap between domain knowledge and image processing algorithms and features. This ontology contains visual concepts used for the visual description of a set of objects of interest.

- An approach for grounding high-level knowledge to image data is also proposed. This can be related to the *symbol grounding* problem which consists of linking meaningfully symbols to sensory information [Harnad, 1990]. These approach is based on both supervised and unsupervised machine learning techniques. This learning phase consists of learning the visual concepts used during knowledge acquisition and not directly the object categories. This means that the object categories are learned through an intermediate level of semantics.

- An object recognition algorithm which makes use of acquired knowledge is also introduced. This algorithm generates hypotheses by using a priori knowledge and verifies these hypotheses in the image to interpret.

One important point is that we want these contributions to lie within the scope of cognitive vision. In particular, we show how the cognitive vision conceptual framework is used to provide an efficient combination of computer vision paradigms, knowledge-based vision and appearance-based vision.

1.3 Dissertation Structure

This manuscript is structured in six chapters.

Chapter 2 introduces the reader to cognitive vision (i.e. scientific foundations, challenges). This emerging discipline is used as a framework for our work which aims at achieving object recognition. We also propose an overview of existing approaches in object recognition: geometric, appearance-based and knowledge-based approaches. Each of these approaches has strengths and weaknesses. Our goal is to use them in a cooperative way in order to make advances in the domain of object recognition. One key element to enable this cooperation is ontological engineering. That is why an overview of this field is also proposed.

Chapter 3 presents our objectives and gives an overview of our approach. As explained in section 1.2, the image interpretation process relies on a priori knowledge which is linked meaningfully with sensory information. We give an overview of our approach which first enables the acquisition of a priori knowledge. The knowledge acquisition bottleneck is reduced by using ontological engineering. Knowledge acquisition results in taxonomy/partonomy of domain object classes described by visual concepts provided by a visual concept ontology. Another important element of our approach is the use of machine learning techniques which are used to ground acquired knowledge with annotated image samples. We also address the object categorization problem by using acquired and grounded a priori knowledge.

Chapter 4 presents the knowledge acquisition phase. Knowledge acquisition is done by interaction with an expert of the application domain. Usually, experts are not skilled in computer vision but are able to produce an accurate visual description of the objects of their domain. Knowledge acquisition consists of achieving the following tasks:

- Domain taxonomy acquisition. This domain knowledge contains both the specialization and part-whole relations between the domain classes. This knowledge which is shared by the specialists of the domain is easy to acquire. This part is independent of vision (visual appearance and image acquisition context).
- Ontology driven visual description of domain object classes which leads to a more detailed symbolic knowledge base. This knowledge is dependent on the image acquisition context (e.g. camera, lightning conditions).

Chapter 5 details how the detection of the visual concepts used during knowledge acquisition for object description is learned. This phase aims at producing a set of visual concept detectors capable of visual concept detection in any image. This can be achieved in three ways by using an image sample database: in a completely *supervised* way by manual segmentation, by using *3-D models*, or in a *weakly-supervised* way by using a combination of supervised and unsupervised machine learning techniques.

Chapter 6 details the categorization phase which uses the system set up by knowledge acquisition and then learning. This chapter details the structure of the requests addressed to the proposed categorization engine. The structure of categorization results is also detailed. The categorization process makes use of the a priori knowledge acquired during knowledge acquisition (i.e. domain classes associated with their visual description). The semantic richness (e.g. specialization and part-whole relations) brought by a priori knowledge enables explicit and detailed categorization. A categorization result is expressed in terms of visual concepts and high-level categories.

Chapter 7 presents some results obtained by applying the proposed approach. This chapter presents the results of a knowledge acquisition process involving palynologists (i.e. *palynology is the study of pollen grains*). Some results on the categorization of texture patches are also shown. The third part of this chapter shows how the proposed approach can be used for semantic image indexation and retrieval purposes in the domain of transport vehicles.

Chapter 8 concludes this manuscript by detailing contributions and also by presenting short-term and long-term perspectives.

Chapter 2

State of the Art

2.1 Towards Cognitive Vision

2.1.1 Definition

Many computer vision systems have never been widely used because of their brittleness and their lack of flexibility. The focus of a lot of researchers in computer vision is now to reduce its brittleness. Cognitive vision takes part of this effort.

A working definition of cognitive vision can be found in [Auer et al., 2005]: *A cognitive vision system can achieve the four levels of generic computer vision functionalities of detection, localization, recognition and understanding. It can engage in purposive goal-directed behavior, adapting to unforeseen changes of the visual environment, and it can anticipate the occurrence of objects and events.*

A cognitive vision system has to exhibit the following faculties: knowing, understanding, reasoning and learning.

One question can be raised when defining cognitive vision: what is the relation between cognitive vision, cognitive systems and computer vision? Answering this question is difficult. One interesting view is given in [Auer et al., 2005]: a cognitive vision system is seen as *visually-enabled cognitive system.* This means that the cognitive vision paradigm places classical computer vision in the context of system-oriented research in cognition.

2.1.2 Scientific Foundations and Functionalities

Cognitive vision has a multi-disciplinary nature. The achievement of cognitive vision implies studying a wide range of disciplines such as computer vision, artificial intelligence, pattern recognition, cognitive science, perceptual psychology or semiotics. Given the current state of our knowledge of cognitive vision, it is too early to ignore any of these fields. All these scientific disciplines have to considered to develop the following functionalities of cognitive vision systems:

1. **Visual Sensing** refers to the mechanism by which the environment impacts the cognitive system.

2. **Architecture** is related to the minimal configuration of a system that is necessary for achieving certain cognitive functionalities. It is also linked to the innate capabilities required for cognitive development.

3. **Memory** is linked to the issue of representation. It can be of different forms: episodic, modal, short-term and long-term.

4. **Learning** refers to process of developing memory. There are many complementary forms of learning which can be used.

5. **Representation** is related to the way knowledge in a cognitive system can be used as support of cognitive capabilities.

6. **Recognition** refers to the ability of discriminating between visual entities (e.g. regions, complicated behaviors). Related functionalities are *detection*, *localization*, *tracking*, *classification* and *categorization*.

7. **Deliberation and Reasoning** refers to the process of using a priori knowledge to derive conclusions in order to solve problems.

8. **Planning** is closely related to reasoning, memory and representation. Planning means dealing with the events of the future (e.g. anticipation, expectation).

9. **Communication** with other systems or agents is a key characteristics of cognitive systems. It has to be effected as a result of cognitive activities. A related functionality is concept formation and visualization.

10. **Action**. Defining action is contentious issue. Does action require physical interaction with the environment? Does a simple change of state constitute an action? This is strongly related to the issue of embodiment, visuo-motor coordination and embodied exploration.

Note that none the issues detailed entail the use of a *unique* paradigm. For instance, reasoning does not necessarily mean symbolic reasoning.

2.1.3 Research Roadmap: Challenges

To achieve cognitive vision, several challenges have to be tackled:

- **Advancement of methods for continuous learning.** Learning in cognitive vision systems is of key importance. To take into account changes in the environment (in the case of the open-world hypothesis), learning capabilities of cognitive vision systems have to be fast, incremental, and continuous.

- **Identification of minimal system architecture(s).** This challenge consists of identifying the minimal set of processing modules and their inter-relationships that enables capabilities such as detection or localization.

- **Goal identification and achievement.** The behavior of cognitive vision systems should be goal-oriented. In some cases, goals can be specified by an external agents. In the emergent cognitive behaviors, what are the goals to achieve?

- **Generalization of operation.** This challenge consists of addressing the problem of transferring skills from one context to another.

- **Utilization and advancement of systems engineering methodologies.** Cognitive systems are expected to have a high degree of complexity. Therefore, their design will make heavy use of software engineering. Several related issues are then raised: how to achieve self-description or self-regulation? What are the relations between software considerations and cognitive considerations?

- **Development of complete systems with well-defined competences.**

- **Creation of research tools**. To ease the development of the discipline, the research community will have to share tools such as physical robotics systems, software development environments, benchmarking scenarios and data.

2.1.4 Existing Cognitive Vision Systems

The **CogVis** project is a European Union funded collaborative project to study the design of cognitive vision systems. Cognitive vision only makes sense in the context of a ”system” where there is a user that provides task information and which uses the information generated by the system.

In the context of CogVis, a cognitive vision system is defined as a system that uses visual information to achieve the following tasks:

1. Recognition and categorization of objects, structures and events.

2. Memory and representation of knowledge.

3. Learning and adaptation.

4. Control and attention.

In this project, task 1 has been achieved by the development of appearance-based vision techniques involving supervised and unsupervised machine learning techniques [Leibe et al., 2004]. Some high-level interpretation techniques have also been developed in order to achieve task 2. These high-level interpretation techniques are presented in [Neumann and Weiss, 2003]. This framework is based on a conceptual model formalized in the $\mathcal{ALCF}(\mathcal{D})$ Description Logic. Both specialization and part-whole relations are expressed in this formalism. Interpretation steps are the following: aggregate instantiation, instance specialization, instance expansion and instance merging. In [Cohn et al., 2003], the incorporation of Qualitative Spatial Reasoning (QSR) into cognitive vision systems is studied. The interpretation problem is modeled as an abduction problem: the system searches for explanations, phrased in terms of the learned spatio-temporal event descriptors, to account for the video data.

Some interesting criteria for evaluating a cognitive vision system have been introduced during the CogVis project:

- The degree of robustness and consistency of the studied methods in natural environments.
- The degree to which these methods by using their specific characteristics, such as spatio-temporal context, non-visual knowledge, learning and multiple cues, improve performance over comparable (or the same) algorithms that does not.
- The degree of scalability with regard to added objects, categories, events and environments.
- The degree of generalization with respect to novel objects, events and environments.

2.2 Object Recognition

Object recognition has been a widely studied research topic for several decades. In this section, we aim at identifying major contributions made in the field of object recognition over the years.

We are interested in the issues associated with generic object recognition: recognizing an object that might never have been observed before, and for which no exact model is available. As explained in [Medioni and François, 2000], the following tasks are required to achieve generic object recognition: description, matching and learning.

- Extraction of descriptions consists of an interpretation of the image data into meaningful entities.
- Matching consists of assigning an identity to the extracted descriptions, a process which involves stored models and comparing them with the image descriptions.
- Learning consists of acquiring objects not previously known to the system and describing similarities and differences with existing objects. This third component is generally ignored in many systems and is performed by the user.

All these tasks rely strongly on the representation scheme used. This scheme affects the choice of the strategies for the description process, the way models are accessed and compared, and the way learning is achieved.

We distinguish three main approaches: geometric methods, knowledge-based methods and appearance-based methods.

2.2.1 Geometric Methods

The first approaches to the object recognition problem found in the literature were geometry-based. Geometric approaches are often synonymous of *model alignment* using geometric invariance [Mundy and Zisserman, 1992]. The matching is achieved by hypothesizing a set of geometric transformations T in order to force the overlapping of the model M on the image I. In this case, the geometric constraints come from the overlapping rate between $T(M)$ and I.

Few geometric approaches really handle the generic 3D object recognition problem. Many geometric approaches are rather focused on the object identification problem (i.e. recognition of an object instance). Nevertheless, contributions made in this field have also been used or extended to handle the generic object recognition problem.

In [Brooks, 1983], Brooks presents a geometry-based system called Acronym. This system constructs a prediction graph representing possible object instances and an observation graph, representing image data and then it attempts to match these graphs. Graph nodes are image ribbons with associated parameter ranges. Graph arcs link adjacent ribbons. Matching is based on a constraint propagation process.

In [Ayache and Faugeras, 1986], the Hyper object recognition system is presented. This system is designed for the recognition of objects lying on a flat surface. The recognition process is structured as a search for consistent set of models and image features. The shape of 2D objects is represented by polygonal approximations of their borders.

In [Ullman and Basri, 1991], a new approach to the alignment problem is proposed. In this paper, it is proved that storing a few views per object obviates the need for maintaining 3D models of objects. All the possible transformation applied to one object are expressed as a linear combination of other views of the same object.

Basri proposes in [Basri, 1996] a method that combines alignment techniques and recognition by prototypes. In this method, objects are divided into classes, where a class contains objects that share a fair number of similar features. Categorization is achieved by aligning the image to prototype objects.

In [Grimson and Huttenlocher, 1990] the generalized Hough transform is used as a method for recognizing objects from noisy data in complex cluttered environments. The generalized Hough transform allows to find arbitrary curves in a given image, without the need for the parametric equation of the curve. This approach consists in constructing a parametric curve description based on simple situations detected during a learning phase. It has been shown that this method does not scale in the case of cluttered scenes or partial occlusion.

In [Belongie et al., 2001],a novel approach to measuring similarity between two shapes and exploiting it for object recognition is presented. The measurement of similarity is preceded by (1) solving for correspondences between points on the two shapes, (2) using the correspondences to estimate an aligning transform. In order to solve the correspondence problem, a descriptor, the shape context, is attached to each point. The shape context at a reference point captures the distribution of the remaining points relative to it, thus offering a globally discriminative characterization.

In [Havaldar et al., 1996], a perceptual grouping hierarchy is used. This approach is inspired by the Gestalt theory. Groups are based on proximity, parallelism, skewed symmetry and closure. Similar groups are grouped into sets. Representation and matching of these sets is done using graphs. The proposed approach enables generic recognition and handles occlusion.

In [Sangineto, 2003], a new approach to object classification based on the idea of geometric abstraction is proposed. A class of objects is described by means of a model which specifies the shape invariants common to all members of the class. A model is a list of geometric constraints fixing the ranges in which local features can vary. An efficient constraint satisfaction algorithm is also proposed.

As explained in [Edelman, 1997], geometry-based methods encounter several computational problems:

- **Need for feature correspondence**. Establishment between models and features (e.g. point, edges, regions) extracted in the image is necessary to achieve object recognition. Therefore, alignment and recognition are highly dependent on the quality of feature detection. It is extremely difficult to detect features in reliable manner. This lack of robustness limits the applications of geometry-based approaches.

- **Lack of abstraction of category information**. A serious problem with alignment-like methods is their too literal treatment of object geometry. Generic object categorization implies abstraction of geometric details. Geometric abstraction can be handled by statistical methods [Shapira and Ullman, 1991]. Nevertheless, the conceptual essence of the objects of interest is difficult to represent with this kind of approach.

- **Lack of an explicit representation of object statistics.** From a statistical standpoint, most alignment methods are designed to treat two objects at a time, instead of capturing several dimensions of variation within a set. Basri deals with this problem in [Basri, 1996].

2.2.2 Knowledge-Based Methods

An excellent overview of knowledge-based vision systems can be found in [Crevier and Lepage, 1997]. One strong ability of knowledge-based systems is to clearly separate knowledge from reasoning. Knowledge-based systems also enables the separation of different types of knowledge: domain knowledge, knowledge about image processing, knowledge about the mapping. Another very interesting property of knowledge-based vision systems is that the interpretation process is explicit (i.e. in terms of concepts of the domain). This explicitness enables user-friendly interaction with the system.

This section gives an overview of some existing knowledge-based vision systems.

Sigma

Matsuyama and Hwang propose knowledge-based system for aerial image understanding called Sigma [Matsuyama and Hwang, 1990]. Knowledge acquisition issues are addressed in this work.

As seen in fig. 2.1, Sigma is structured in four main modules:

1. The Geometric Reasoning Expert (GRE). GRE is the central reasoning module in the system: it constructs the description of the scene by establishing spatial relations between objects. This module uses the world model which is formalized by frames. The world model is used to generate hypotheses which have to be verified in the image. This process is called the evidence accumulation process. The Iconic/Symbolic database is used to store consistent pieces of evidence. Bottom-up and top-down analyzes are integrated into a unified reasoning process.

2. The Model Selection Expert (MSE) reasons about the most promising appearance of the target object to be detected in the top-down analysis. This module is dedicated to the mapping between high-level knowledge and low-level image data.

3. The Low Level Vision Expert (LLVE). LLVE is dedicated to low level image processing tasks: segmentation and feature extraction.

4. The Question and Answer Module (QAM). QAM is used to retrieve information during the interpretation process.

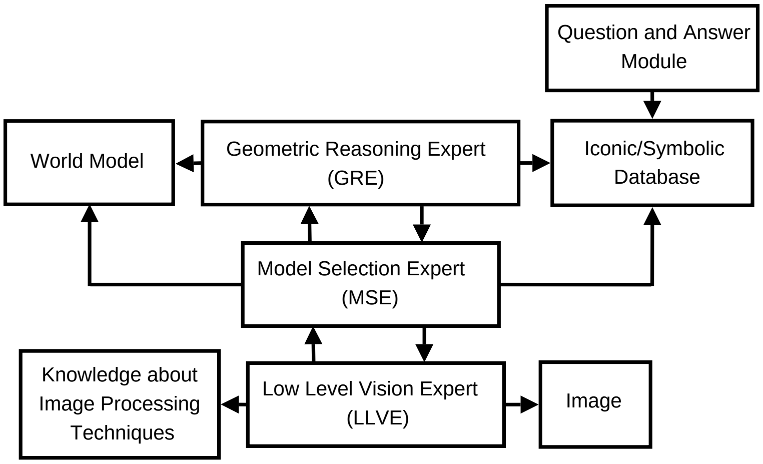

Question and Answer Module
World Model
Geometric Reasoning Expert (GRE)
Iconic/Symbolic Database
Model Selection Expert (MSE)
Knowledge about Image Processing Techniques
Low Level Vision Expert (LLVE)
Image

Figure 2.1: Architecture of the Sigma system. Sigma is designed as a distributed architecture. Modules communicate by query/answer mechanisms.

To the best of our knowledge, this system has only been used for aerial image understanding purposes.

Schema

In [Draper et al., 1989], Draper et al. present a knowledge-based vision system used for outdoor scene image interpretation.

The Schema system is structured in three levels:

- The high-level contains a semantic network of schemas. Each schema is specialized for the identification of one particular class of object. A schema models the appearance of an object and contains the strategy (i.e. invocation of low-level and intermediate-level routines) used for recognition.
- The role of the low-level is to extract image primitives (i.e. contours and regions) and to produce tokens containing attributes. Attributes are computed for tokens. For instance, the shape of a region is described by its compactness.
- The intermediate level role is to form and verify hypotheses about objects in the scene. It provides functionalities such as: perceptual grouping, geometric model matching, token relations management, and knowledge-directed segmentation.

The Schema system has been designed to run in a parallel environment and is able to deal with several hypotheses simultaneously.

Ocapi/Classic: a cooperative approach for natural complex object recognition.

In [Ossola et al., 1996], an approach based on the cooperation between two knowledge-based systems is presented. In this approach, the problem of complex object recognition is divided in two distinct sub-problems: high-level semantic interpretation and low-level image processing. A dedicated knowledge-based system is associated with each sub-problem. This work emphasizes on communication issues between the two knowledge-based systems. An overview of this approach is sketched in fig. 2.2.

The approach used to cope with the low-level image processing problem is program supervision ([Clement and Thonnat, 1993], [Shekhar et al., 1998]). A priori knowledge knowledge on a library of programs improves the use of these programs. As seen in fig. 2.3, the following mechanisms are involved in program supervision: *planning*, *execution*, *evaluation*, *repair*.

A related approach based for the automatic generation of image processing applications can be found in [Clouard et al., 1999]. It is based on hierarchical, opportunistic and incremental planning by means of knowledge sources of the Blackboard model, which enables taking into account planning, evaluation and knowledge acquisition issues.

ERNEST

In [Niemann et al., 1990], a system environment for the treatment of general problems of image and speech understanding is presented. A framework for the representation

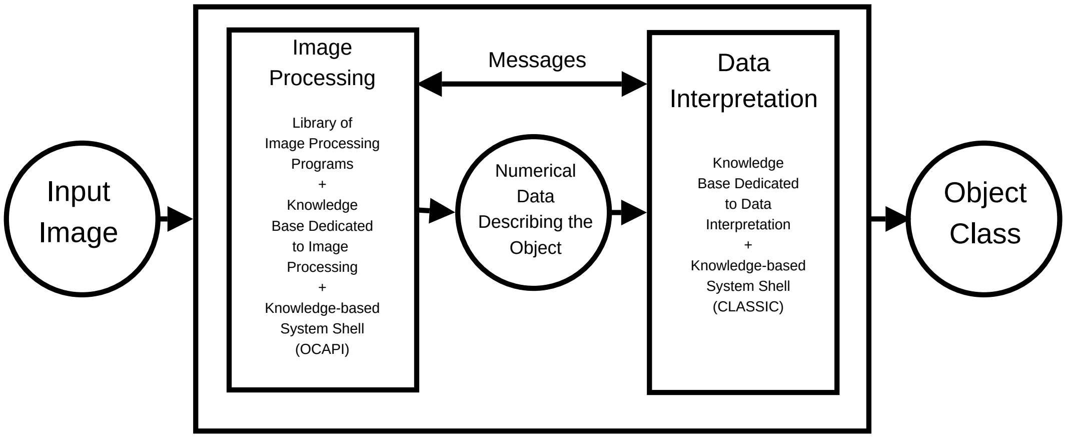

Input Image
Image Processing
Library of Image Processing Programs + Knowledge Base Dedicated to Image Processing + Knowledge-based System Shell (OCAPI)
Messages
Numerical Data Describing the Object
Data Interpretation
Knowledge Base Dedicated to Data Interpretation + Knowledge-based System Shell (CLASSIC)
Object Class

Figure 2.2: A cooperative approach for natural complex object recognition. The input of the system is an image containing an object to classify. The output is the name of the class of the object.

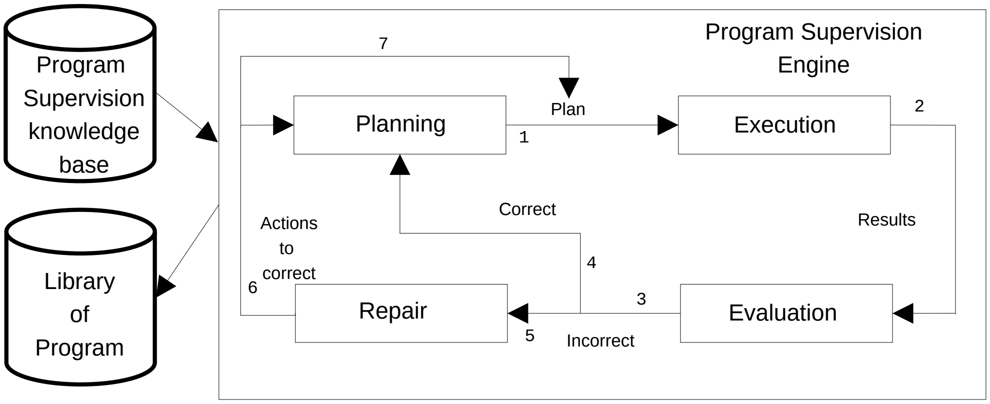


Figure 2.3: An overview of program supervision

of declarative and procedural knowledge based on *semantic networks* is proposed. Two different image understanding systems have been developed by using this framework: a scintigraphic image analysis system and a industrial scene analysis system. The problem of object representation and the problem of generic control are addressed in this work.

One strong criticism made by Draper et al. in [Draper et al., 1996] is that most knowledge-based systems are tailored to one application domain. The close-world assumption entailed by a priori knowledge modeling is also often criticized. Another weakness of knowledge-based systems is that the use of explicit knowledge is not really suited for modeling the variability, the changes and the complexity of the world.

Draper also emphasizes on the knowledge acquisition bottleneck: it is hard to scale knowledge-based approaches up to large problem. On the other hand, the semantic richness of knowledge-based approaches is often synonymous of user-friendliness for the end-users of the systems built by using these approaches.

2.2.3 Appearance-based Methods

Appearance-based approaches do not rely on explicit 3D object models or an explicit a priori knowledge but use multiple views. In the appearance-based paradigm, objects are modeled by a set of images. The model set consists of the original images, considered as feature vectors.

In [Swain and Ballard, 1991] an object is represented by its color histogram. Objects are identified by matching a color histogram from an image region with a color histogram from a sample object. The matching is performed by histogram intersection. The method is robust to changes in the orientation, scale, partial occlusion and changes of the viewing position. One drawback of the method is its sensitivity to lighting conditions.

In [Schiele and Crowley, 2000], a generalization of this approach by introducing multidimensional receptive field histograms to approximate the probability density function of local appearance is proposed. The recognition algorithm calculates probabilities for the presence of objects based on a small number of vectors of local neighborhood operators such as Gaussian derivatives at different scales.

In [Schmid and Mohr, 1997], an object recognition approach based on the combination of differential invariants computed at key points with a robust voting algorithm and semi local constraints is proposed. The recognition is based on the computation of similarity (represented by the Mahanalobis distance) between two invariant vectors. Matching is performed on discriminant points of an image and a standard voting algorithm is used to find the closest model to an image. An overview of scale and affine invariant point detectors can be found in [Mikolajczyk and Schmid, 2004].

A model to represent objects as probabilistic constellations of rigid parts is proposed in [Weber et al., 2000]. The variability within a class is represented by a joint probability density function on the shape of the constellation and the appearance of the parts. The method automatically identifies distinctive parts in the training set. The model parameters are trained by expectation-maximization.

In [Fergus et al., 2003], the approach presented in [Weber et al., 2000] is extended. A method to learn and recognize object class models from unlabeled and unsegmented cluttered scenes in a scale invariant manner is also presented. Objects are modeled as flexible constellations of parts. A probabilistic representation is used for all aspects of the object: shape, appearance, occlusion and relative scale. An entropy-based feature detector is used to select regions and their scale withing the image. Learning consists of estimation of the parameters of the scale-invariant object model. The recognition step uses this model in a Bayesian manner to classify images. Several improvements are made on [Weber et al., 2000]: explicit model of variability of appearance, simultaneous learning of shape and appearance and efficient learning of new categories.

In [Fei-Fei et al., 2004], a method for learning object categories from just a few training images is presented. It is quick and it uses prior information in a principled way. The approach is tested on a data set composed of images of objects belonging to 101 widely varied categories. The proposed method is based on making use of prior information, assembled from (unrelated) object categories which were previously learned. A generative probabilistic model is used, which represents the shape and appearance of a constellation of features belonging to the object. The parameters of the model are learned incrementally in a Bayesian manner. Incremental learning is fast, making real-time learning feasible.

In [Csurka et al., 2004] SIFT [1] descriptors [Lowe, 2004] are used to build bags of key points used to achieve generic visual categorization. A bag of key points is based on vector quantization of SIFT descriptors. Two types of classifiers are used: Naive Bayes and Support Vector Machines. The SVM classifier outperforms the Naive Bayes classifier. The method is robust to background clutter and produces good categorization accuracy. The approach has been evaluated deeply and results are presented as a confusion matrix.

An object categorization method in real-world scenes is presented in [Leibe et al., 2004]. This approach is a combination of recognition and segmentation into one process. The proposed method is based on Harris interest point detector [Harris and Stephens, 1988] which are used to extract images patches. Extracted image patches are then compared to a pre-defined code book built form sample images. This code book is used as a probabilistic voting space during recognition. Patch selection is finally refined in order to obtain a category-specific segmentation. One strong point of this approach is that it can deal with several object instances in the image.

Edelman exposes computational problems of appearance-based methods in [Edelman, 1997]:

- **Combining diagnosticity with invariance**. The main problem of feature-space methods is finding features that provide reliable discrimination among similar objects, along with invariance across object transformations. This is related to the stability versus sensitivity issue mentioned by Marr [Marr, 1982].

- **Difficulty of learning from examples in multidimensional spaces**. Appearance based techniques are often synonymous with learning from examples in high-dimensional space (e.g. histogram). This leads to the problem known as the curse of dimensionality: the exponential dependence of the required number of examples on the number of dimension of the representation space. Dimensionality reduction is then of strong importance.

It can be added that this kind of approach is not suited to applications where the number of image samples is low. Experience also shows that the image samples have to be well-chosen for obtaining efficiency. An interesting challenge is to integrate these techniques in a conceptual framework (which takes into account domain knowledge) suited, for instance, to end-user interaction.

[1] Scale Invariant Feature Transform

2.2.4 Conclusion

We have exposed three categories of techniques used for solving the difficult problem of object recognition. Geometric methods have been used mainly for object identification purposes. The generalization of these techniques to the recognition of natural objects is difficult. Knowledge-based approaches have been used for object categorization. One strength of this kind of approach is the semantic richness which enables user-friendly interaction with the end-users. Appearance-based techniques are efficient for object recognition problems provided that enough samples are available. One real challenge is to combine the best of these two approaches. Such a combination can be obtained by the use of a conceptual framework relying on ontological engineering and knowledge representation techniques.

2.3 Ontological Engineering and Knowledge Representation

2.3.1 Ontological Engineering

The term *Ontology* refers to the philosophical discipline which deals with the nature and the organization of reality. In this sense *Ontology* tries to answer the question: *what is being?*, or *what are the features common to all beings?*. The meaning of this term has slightly evolved in the artificial intelligence community. Gruber defines the notion of ontology in [Gruber, 1993]: *An ontology is an explicit specification of a conceptualization.*

In [Guarino and Giaretta, 1995] several complementary definitions are given.

A *conceptualization* is defined as *an intensional semantic structure which encodes the rules constraining the structure of a piece of reality.*

Ontological engineering is defined as *the branch of knowledge engineering which exploits the principals of Ontology to build ontologies.*

In [Gandon, 2002], a *concept* is defined as *a notion usually expressed by a term (or more generally by a sign). A concept represents a group of objects or beings sharing characteristics that enable us to recognize them as forming and belonging to this group.*

As explained in [Bachimont, 2000] the aim of ontologies is to define which primitives, with their associated semantics, are necessary for knowledge representation in a given context.

An ontology is composed of several entities:

- a set of concepts (C) (e.g. geometric concepts)
- a set of relations(R) (e.g. spatial relations)
- a set of axioms (e.g. transitivity, reflexivity, symmetry of relations)

Two partial orders $\preceq_C$ and $\preceq_R$ define the concept hierarchy and the relation hierarchy, respectively. An ontology is supposed to be the support of reasoning mechanisms.

To be efficient, communication between people and software systems must rely on a shared understanding. As explained in [Gandon, 2002], lack of shared understanding leads to difficulties in identifying requirements and to limited inter-operability or reusability.

A relevant example of ambiguity was given by Gómez-Pérez. What should be answered to the question ”What is a pipe?”. There are several possible answers: a short narrow tube with a small container at one end, used for smoking tobacco; a long tube made of plastic or metal that is used to carry water or oil or gas; a temporary section of computer memory that can link two different computer processes.

These problems are often met when building or interacting with computer vision systems. Ontologies are a common base to build on and a shared reference to align with [Gandon, 2002]. This shared reference is obtained by a consensus called ontological commitment. That is why ontological engineering can be useful for the cognitive vision community.

As explained in [Blazquez et al., 1998], ontology development process has to be done in four distinct phases.

- the first one is called *specification* and states why the ontology is built and who are the end-users
- the next phase is *conceptualization* and leads to a structured domain knowledge
- then comes the *formalization* phase that transforms the conceptual model into a formal model
- finally, *implementation* transforms the formal model into a computational model

The ontology life-cycle is depicted in fig. 2.4. This methodology has been used to design the visual concept ontology presented in section 4.3.

As explained in [Gandon, 2002], the ontology life-cycle is very important because it impacts what is built on this ontology. For instance, knowledge base coherence has to be maintained. Changes in the formalization may have impact on the engine that reasons on the ontology. Ontologies have to be managed with care.

2.3.2 Knowledge Representation

One important step of the ontology life-cycle is the formalization step (see fig. 2.4). This section aims at giving an overview of the existing knowledge representation languages which can be used for ontology formalization. There exists several knowledge formalization languages: first order and proposition logics, semantic networks, conceptual graphs, frame and description logics. A formalism provides a symbolic system (e.g. syntax, axioms, inference rules) and the semantics attached to it. When formalizing an ontology, it is important to find a formalism which provides adequate primitives to capture the aspects of the ontology. A view of knowledge representation languages can be found in [Corcho and Gómez-Pérez, 2000].

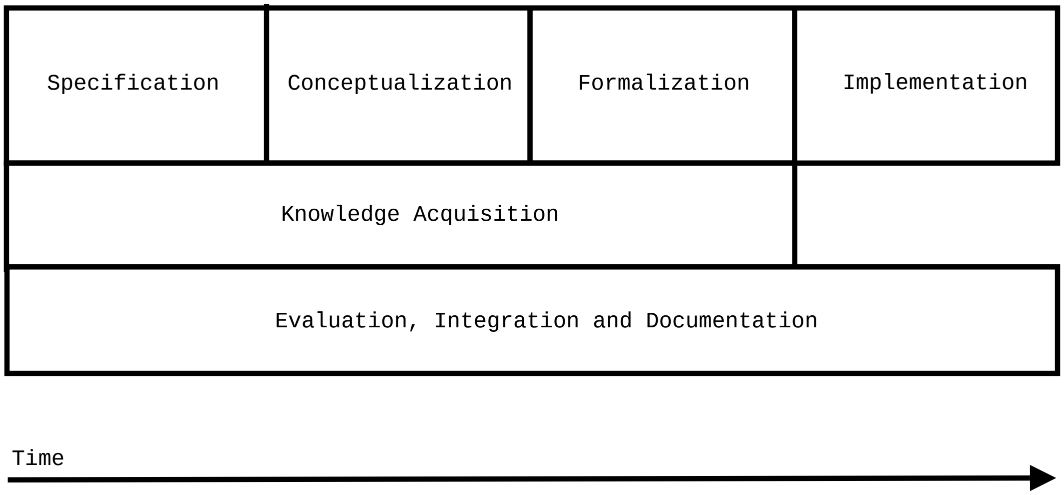


Figure 2.4: The ontology life cycle is composed of several phases. These phases are supported by activities such as *evaluation*, *integration*, *documentation* and *knowledge acquisition*.

Logic of Propositions

Propositional logic originates from philosophy and is the foundation knowledge formalization language. Propositional logic expressiveness is limited: it only considers relations between propositions without considering the structure and the nature of the proposition. For instance, it does not enable to represent the difference between individuals and categories.

First Order Logic

First order logic (FOL) includes propositional logic. The addition of existential and universal quantifiers enable to differentiate individuals from categories. For instance, FOL enables to write the following statement: $\forall x\ dolphin(x) \supset mammal(x)$. This means that every dolphin is a mammal. This is a subsumption relation which has the semantics of set inclusion. The Cyc ontology [2] (which aims at being an ontology of common sense) is formalized in FOL.

Semantic Networks

Semantic networks were introduced by Quillian [Quillian, 1985]. Semantic networks are graphs which represent concepts and their relationships to each other. Graph nodes represent concept and graph arcs represent relations between concepts. When using semantic nets, several problems can be encountered.

The unpredictability of the inference process makes reasoning difficult to debug. Semantic nets are relatively under-constrained. This entails a large number of possible rep-

[2] http://www.cyc.com

resentations of the same situation. Moreover, knowledge bases represented in the semantic nets formalism often seem to be disorganized. The reasoning methods for a semantic net system have to be specified for each possible interaction of arcs. Whereas a logic-based representation usually has a very small number of powerful inference techniques available, a semantic net system usually has a large number of special purpose inference methods. To answer queries, especially queries with negative answers, it is often necessary to search most or all of the semantic net. Heuristic methods to reduce the size of the search have been proposed, but have not been particularly successful.

Conceptual Graphs

Conceptual Graphs (CG) are inspired by semantic networks [Sowa, 1984]. The main improvement on semantic networks is that they rely on a formal logical layer. Conceptual Graphs enable friendly presentation of logic to human. Conceptual graphs have several representations:

- DF (Display Form): a graphical representation;
- LF (Linear Form): a textual representation equivalent to the DF;
- CGIF (Conceptual Graph Interchange Format): for transmission between systems.

A CG is a bipartite oriented graph. There are two types of nodes in the graph: concept nodes and relation nodes. The arcs are oriented and always link a concept node to a relation node. CGs are existential and conjunctive statements. The arity of relations is an integer n which represents the number of concepts they can be linked to. Concepts and primitives have a type which can be a primitive type or a defined type. The ontological knowledge upon which CGs are built is represented by the support, made of two subsumption hierarchies structuring concept types and relation types. This is the terminological level. Relations have a fixed arity and a signature (i.e. the types of the concepts linked by the relation).

The core reasoning operator of Conceptual Graphs is the computation of subsumption relations between graphs. It is based on the notion of projection, a graph homomorphism such that a graph G subsumes a graph G' iff there exists a projection from G to G'. The projection takes into account specialization of relations and concepts. This means that nodes of the graph G must be of the same type or must be subsumers of the nodes in G' they are mapped to. The logical definition of a conceptual graph G is usually noted $\Phi(G)$.

An example of conceptual graph is shown in fig. 2.5.

Frames

The notion of Frame was introduced by Minsky in the mid seventies [Minsky, 1975]. In the frame formalism, frames are organized hierarchically. This hierarchy is based on a specialization relation: a-kind-of. This relation is a partial order and is reflexive, transitive

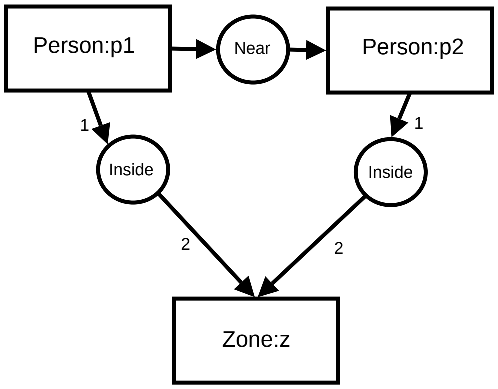


Figure 2.5: This conceptual graph states that a person $p1$ is near a person $p2$ and that both of them are inside a zone z.

and anti-symmetric. From a structural point of view, this means that all the descendants of a class own all the attributes and methods of the inherited class. From a conceptual point of view, an instance of a subclass C is also an instance of superclass(es) of C. A frame is composed of attributes and facets. Facets can be declarative or procedural to specify the nature (type, domain, cardinality, value) or the behavior of an attribute (default value, procedures to calculate the value, filters). Point of views can be defined to build different hierarchies of classes capturing different conceptualization while enabling an object to take into account all the aspects of its class in the different views.

Description Logics

Description Logics (DL) [Baader et al., 2003] are based on predicate logics, semantic networks and frame languages. Two types of knowledge are distinguished: terminological knowledge where concepts and roles are represented and manipulated and assertional knowledge where assertions and manipulations about individuals are made. The assertional level is usually called the A-Box and the terminological knowledge is called the T-Box. A T-Box is composed of a set of concepts being either primitive or defined by a term, a set of roles, or a set of individuals. A concept is a generic entity of an application domain representing a set of individuals. An individual is a particular entity, an instance of a concept. A role is a binary relation between individuals. A role can be primitive or defined. Concepts are organized in a subsumption hierarchy. There are several description logics. Each DL is associated with a set of predefined constructors (Table 2.1).

For example, the concept of ”individuals having a female child” is expressed as $\exists hasChild.Female$. The concept of ”individuals all of whose children are female” is expressed as $\forall hasChild.Female$. The concept $(\geq 3\ hasChild) \cap (\leq 2\ hasFemaleRelative)$ represents the ”individuals having at least three children and at most two female relatives”.

As explained in [Möller et al., 1999], typical inference services offered by a DL system (e.g. Racer [Haarslev and Möller, 2001], Fact) [Horrocks, 1998] are *subsump-*

DL Name	Available constructors	Comment
$\mathcal{AL}$	$\{\top, \bot, \neg A, C \cap D, \forall r.C, \exists r\}$	A is a primitive concept C,D are defined concepts and r is a role $\top$ is the top concept and $\bot$ is the empty concept
$\mathcal{FL}$	$\{C \cap D, \forall r.C, \exists r, r\|C\}$	concept negation and role restriction
$\mathcal{ALN}$	$\mathcal{AL} \cup \{\geq n\ r, \leq n\ r\}$	at-least and at-most cardinality
$\mathcal{ALR}$	$\mathcal{AL} \cup \{r1 \cap r2\}$	conjunction of roles

Table 2.1: Some Description Logics. More expressive DLs can be created by combining existing DLs. This is often at the cost of efficiency.

tion check, *consistency check*, *classification* and *abstraction*. The following languages are based on DLs: Loom [MacGregor, 1991], Classic [Borgida et al., 1989] or Owl [McGuinness and van Harmelen, 2004].

2.4 Conclusion

The weaknesses of classical computer vision systems have led to the emergence of a new discipline called *cognitive vision*. We have presented the scientific foundations of this discipline. We have also listed the functional capabilities that a cognitive vision system should be able to achieve. The cognitive vision community has identified several challenges which will have to be tackled to obtain complete (i.e. with the capability of learning, knowing, understanding and reasoning) cognitive vision systems.

One very interesting aspect of cognitive vision is that it is a system-oriented approach. It can be seen as a conceptual framework (i.e. functionalities, foundations) which does not commit to any particular paradigm (e.g. symbolic reasoning and representation). This is a fundamental difference with classical computer vision which tends to oppose different paradigms.

We are interested in the problem of image interpretation and more precisely in the problem of object class recognition. Our goal is to address this problem by taking into account the landmarks set by the cognitive vision community.

This difficult problem has been tackled by different approaches: geometric, appearance-based and knowledge-based approaches. Most geometric-based object recognition methods are not really efficient for non manufactured objects: the extraction of geometric primitives is still an open problem. Knowledge-based approaches have a high-degree of explicitness and enable a clear separation between the different underlying problems of image interpretation (i.e. image processing, mapping, high-level interpretation). Their most important weakness is that knowledge acquisition is difficult. Many important recent contributions have recently been made in appearance-based vision ([Schmid and Mohr, 1997], [Lowe, 2004]). These techniques have led to efficient object class recognition systems. The weakness of this category of approaches is their lack of explicitness and the fact they are currently not well integrated with high-level semantic knowledge. For instance, such an integration is very important in the application domain of image indexing and retrieval. From a general point of view, the high-level knowledge layer is of key importance when interacting with the user.

Each of these paradigms has strengths and weaknesses. We believe that the framework of cognitive vision is well-suited for taking the best of each of these approaches.

We believe that a key element of this unification inside a cognitive vision system are the alignment and the shared understanding brought by ontological engineering. Several knowledge representation languages can be used for formalizing ontologies. The choice of a knowledge representation language has to be done by taking into account the expressivness of the language, the efficiency of the reasoning mechanisms supported by the knowledge representation language, and the ease of use of the language.

Chapter 3

Approach Overview

3.1 Introduction

This thesis forms part of the semantic image interpretation activities of the Orion team. In [Liu et al., 1994], a knowledge-based system is used for the classification of planktonic foraminifera. In [Ossola et al., 1996], a distributed architecture for object recognition is proposed. In [Hudelot, 2005], this distributed architecture is extended by a visual data management module, a knowledge-based system dedicated to the mapping of high-level knowledge with low-level image processing algorithms and features and also to spatial reasoning. We aim at improving on the past contributions of the team while keeping in mind the framework of cognitive vision.

Our goal is to propose an approach that enables an expert of a given application domain (e.g. biologist, astronomer) to build a dedicated and operational *object categorization system* without requiring image processing knowledge. The resulting system has to exhibit faculties of *knowing*, *understanding*, *reasoning*, and *learning*. Moreover, we aim at proposing an approach with properties of *re-usability* and of *convenience*. This is a difficult challenge.

We have seen in chapter 2 that the difficult problem of object recognition has been tackled by many different approaches. Our goal is to propose a framework that takes the best of each approach. In particular, we believe that the semantic richness brought by knowledge-based techniques and the efficiency of appearance-based techniques can be combined.

The image interpretation problem can be decomposed in three sub-problems: *high-level interpretation*, *mapping* and *image processing*. This decomposition of the image interpretation problem is inspired by the paradigm of Marr [Marr, 1982]. This view in different levels of abstraction of the image interpretation problem is used to structure the way we cope with the object categorization problem.

This chapter provides details on our *objectives* in section 3.2 and gives an overview of the proposed approach in section 3.3. The proposed approach is composed of three phases: *a knowledge acquisition phase*, *a visual concept learning phase*, and *an object categorization* *phase*. The knowledge acquisition phase results in the acquisition of the knowledge used as the support of high-level interpretation but also in the partial reduction of the *semantic gap* between high-level knowledge and low-level image processing features and algorithms. The role of the visual concept learning phase is to completely fill this gap. The resulting object categorization system is fully *automatic* and can perform object categorization tasks. Our main contributions are related to the high-level interpretation level and to the mapping level of the image interpretation problem.

3.2 Objectives

3.2.1 An Approach for Building Object Categorization Systems

As seen in fig. 3.1, the goal of this work is to propose an approach that enables an expert to set up dedicated object categorization systems. The resulting operational object categorization system is fully automatic and can be used for performing object categorization tasks (fig. 3.2). Building operational and dedicated object categorization systems is a difficult task. The problem defined here is at a higher level of abstraction than the problem of building a tailored object categorization system.

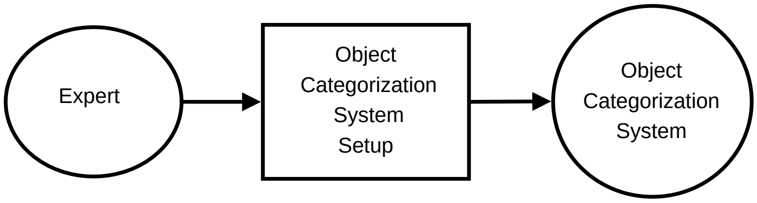

Expert
Object Categorization System Setup
Object Categorization System

Figure 3.1: Our goal is to propose an approach that enables an expert to produce an object categorization system dedicated to his object of interest.

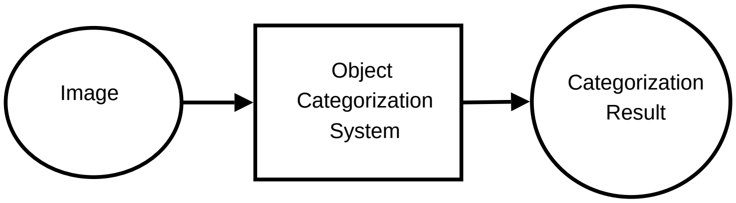


Figure 3.2: Once the object categorization has been set up, it can take images as an input to produce categorization results as an output.

3.2.2 Properties of the Approach

This thesis aims at proposing an approach for building object categorization systems which has properties of re-usability and of convenience.

Re-usability

The first level of re-usability is at the level of the domain of interest. One goal of this work is to propose an approach which can be used for building object categorization systems in different application domains. The main idea is to propose to an expert of any domain of interest, a set of components that enables him/her to build a dedicated object categorization system. A certain level of *performance* and of *robustness* also has to be obtained.

Another level of re-usability is from the point of view of the computer vision specialist. This means that the approach has to rely on well-defined components which solve tractable problems. It is also important to clearly define the communication layers between the different components.

Another important requirement is to ensure that our approach can be *extended*. For the expert of the domain, this means having the possibility to introduce new categories of objects or to improve the performance of the object categorization system. For the computer vision specialist, this means having the possibility to extend the approach easily (e.g. adding new image processing algorithms).

Convenience

Another important property that we are aiming is the property of convenience. By convenience, we mean that the set up or the extension of an object categorization system has to be done by user-friendly *interactivity*. In particular, it is important that the interactive set up by an expert does not require to manipulate image processing notions: the complexity of image processing has to be hidden.

Another facet this property of convenience is the notion of *autonomy*. Once the object categorization is set up, it must be able to perform categorization tasks completely automatically.

3.2.3 Conclusion

Our goal is to propose an approach that enables an expert of a domain of interest to build an object categorization. The objective is to propose an approach with properties of *re-usability* and of *convenience*. These properties are important for the computer vision specialist and also for the domain expert.

3.3 Proposed Approach

We propose an approach to solve the problem presented in the previous section that takes into account the paradigm of Marr [Marr, 1982]. We make a clear distinction between the three levels of abstraction of the image understanding problem: the high-level interpretation problem, the mapping problem, and the image processing problem. Our contributions are mainly related to the high-level interpretation problem and to the mapping problem. Our approach is composed of three main phases:

Phase one consists of acquiring high-level knowledge (i.e. a hierarchy of object classes with their subparts) used for semantic interpretation. This phase also consists of acquiring the visual description of the objects of interest in terms of visual concepts. The role of these visual concepts is to reduce the gap between high-level knowledge and low-level features. This phase is related to the high-level interpretation problem and also to the mapping problem.

Phase two is dedicated to the learning of the mapping between image data extracted by image processing and high-level knowledge. This is done by exploiting the visual description of the objects of interest and a set of annotated and segmented image samples. This phase is related to the mapping problem.

Phase three consists of using the results of knowledge acquisition for object categorization purposes.

3.3.1 Decomposition of the Semantic Image Interpretation problem

The complex problem of semantic image interpretation is divided into three more tractable sub-problems (fig. 3.3). This decomposition is directly inspired by the paradigm of Marr [Marr, 1982].

1. **The image processing problem** consists of extracting image data (e.g. regions) from the image. It also consists of computing their numerical description. All the processes involved remain at a low-level (e.g. region segmentation, histogram computation, smoothing).

2. **The mapping problem** consists of finding a correspondence between the image data extracted by image processing (e.g. regions, edges) and the high-level concepts of a domain of interest. The mapping problem involves complex processes such as spatial reasoning or uncertainty management. This layer is at an intermediate level of semantics (e.g. a pink circular surface).

3. **The semantic image interpretation problem** consists of achieving the understanding of a scene in terms the concepts of the domain (e.g. a Poeceae pollen grain, an aircraft).

Figure 3.4 illustrates the processes involved at each level of abstraction during the interpretation of an image.

3.3.2 Knowledge Acquisition

First comes the knowledge acquisition phase which is done by interaction with an expert of the application domain. Usually, experts are not skilled in computer vision but are able

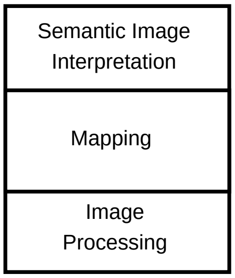


Figure 3.3: The problem of semantic image interpretation can be divided into three more tractable sub-problems

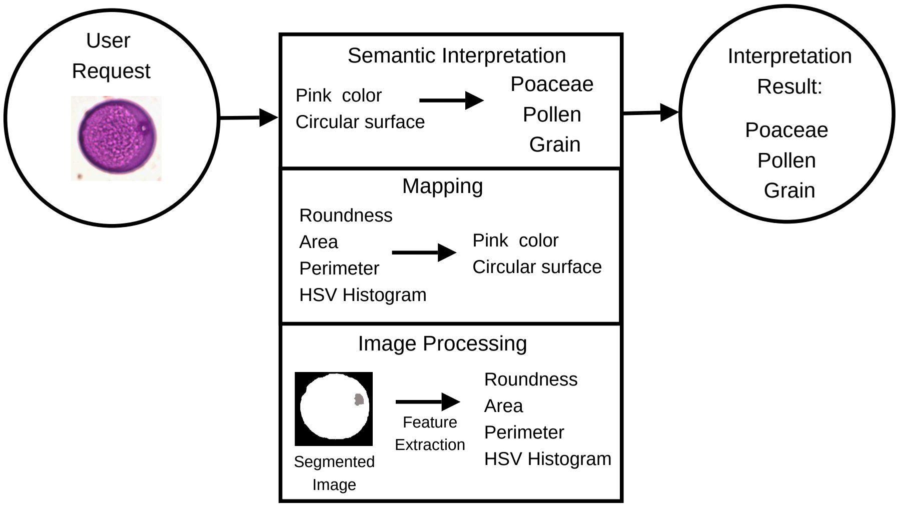


Figure 3.4: From a user request to a semantic interpretation. Low level image processing leads to a segmented regions and to their numerical description. At the intermediate level, a symbolic abstraction is derived from the extracted features (e.g. Pink, Circular). High-level interpretation is made in the terms of the concepts of domain (e.g. Poaeceae)

to produce an accurate visual description of the objects of their domain. Our goal is to use this visual description to reduce the semantic gap between high-level knowledge and low-level features.

The knowledge acquisition phase consists of performing the following tasks:

- **Domain taxonomy acquisition**. This domain knowledge contains both the specialization and part-whole relations between the domain classes. This knowledge which is shared by the specialists of the domain is easy to acquire. This part is independent of vision (visual appearance and image acquisition context).

- **Ontology driven visual description** of each class which leads to a knowledge base structured as a taxonomy/partonomy of domain classes described by visual concepts. This knowledge is dependent of the image acquisition context (e.g. camera, lighting conditions).

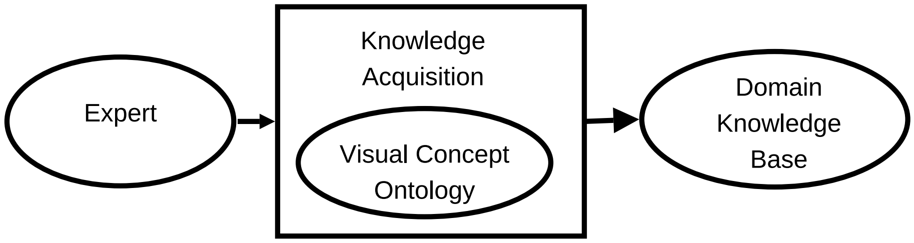


Figure 3.5: Knowledge Acquisition Phase Overview. The expert is involved in a knowledge acquisition process that results in domain knowledge base structured as taxononmy/partonomy of object classes described by visual concepts. The visual concept based description *partially* fills the semantic gap between high-level knowledge and low-level features and algorithms in a user-friendly way.

As seen in fig. 3.5, the knowledge acquisition process leads to a knowledge base in which a set of domain classes are described by visual concepts. High-level knowledge acquisition result is a structured knowledge-base with a well-defined semantics that can support reasoning mechanisms. Another important result of this knowledge acquisition process is the reduction of the *semantic gap*. Indeed, the visual concepts involved in the description of the object classes are close to low-level image features. Thus, the mapping problem is simplified.

As shown in fig. 3.5, one key element of the knowledge acquisition process is a *visual concept ontology*. This ontology is composed of different categories of visual concepts: color visual concepts, texture visual concepts, and spatial visual concepts. These concepts can be used to describe the visual properties of the object classes of interest (e.g. *Hue*, *Brightness*, *Saturation* for color visual concepts). The complete ontology is composed of 144 visual concepts (e.g. *Granulated* Texture, *Coarse* Texture, *Circular* Surface, *Dark*, *Elongated*, *Small*, *Circular*, *Pink*). A sub-hierarchy of the visual concept ontology is given in fig. 4.4. The depth of the ontological tree is 8. This ontology can be extended and can be specialized depending on the application domain. Numerical features are associated with visual concepts and define how visual concepts are computed on image data. Examples of numerical features associated with visual concept are: color coherence vectors [Pass et al., 1996] for characterizing the hue of the objects of interest; co-occurrence matrices [Zhang and Tan, 2002] for characterizing the pattern of the objects of interest; SIFT features ([Lowe, 1999], [Csurka et al., 2004]) for characterizing the geometry of the objects of interest.

3.3.3 Visual Concept Learning

During the knowledge acquisition process, expert knowledge has been acquired. The role of visual concept learning is to fill the gap between symbols used during knowledge acquisition and low-level image features. Segmented and annotated image samples of domain objects are used for that purpose. Sample annotation consists of labeling a set of segmented images (fig. 3.6) by visual concept names. This means that each segmented region is associated with one or several visual concepts. This process can be done manually by involving the expert in a manually or weakly-supervised approach which uses an input image training set. The complementary use of 3-D models is also possible (fig. 3.7).

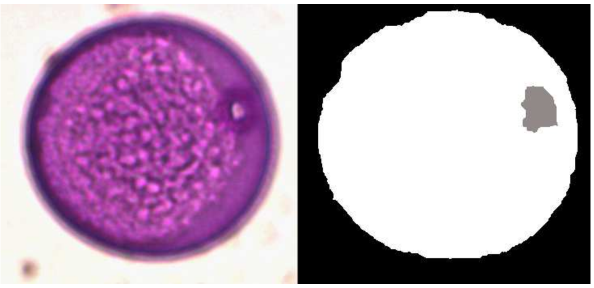


Figure 3.6: Image of a pollen grain on the left. On the right, manual segmentation associated with the image. Three different regions are considered: the background in *black*, the pollen grain in *white* and one of its subparts in *grey*. To enable learning, each region is annotated with visual concepts (e.g. *Pink*, *Circular*, *Big* for the pollen grain.)

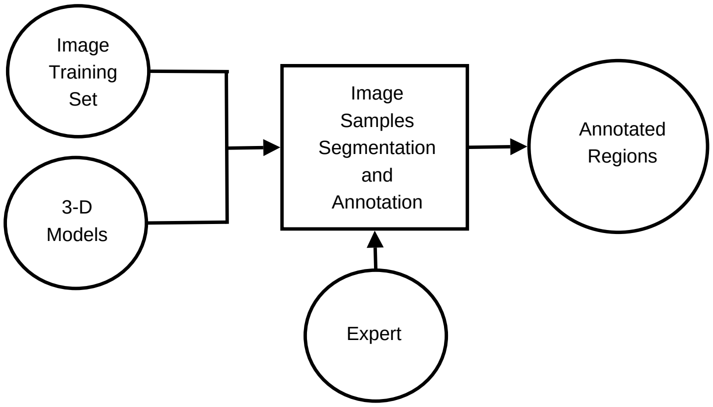


Figure 3.7: One important element of the visual concept learning phase is the image samples segmentation and annotation process. This can be achieved by involving the expert in a manually or weakly-supervised approach which uses the input image training set. The use of 3-D models is also possible.

As seen in fig. 3.8, three tasks are involved in the object learning process.

1. Object learning consists of learning how to detect the visual concepts describing the classes of the domain taxonomy. For specific applications, some classes are not relevant. Therefore, it may be needed to restrict learning process to a subpart of the whole domain knowledge. That is why only a subpart of the domain taxonomy can be considered. Object learning uses a set of regions of interest annotated by visual concepts. The output of object learning is a set of visual concept detectors capable of visual concept detection in any image.

2. Visual concept learning consists of training a set of visual concept detectors by using the features extracted by feature extraction on the annotated regions. These visual concept detectors are trained to the recognition of visual concepts used for the description of the domain classes. A visual concept detector is dedicated to the detection of a visual concept and is obtained by training a support vector machine (SVM) with *positive* and *negative* samples of this visual concept.

3. Feature extraction consists of extracting the numerical features associated with the visual concepts on the annotated regions of interest (e.g. extracting color histograms for learning how to detect the visual concept *Blue*).

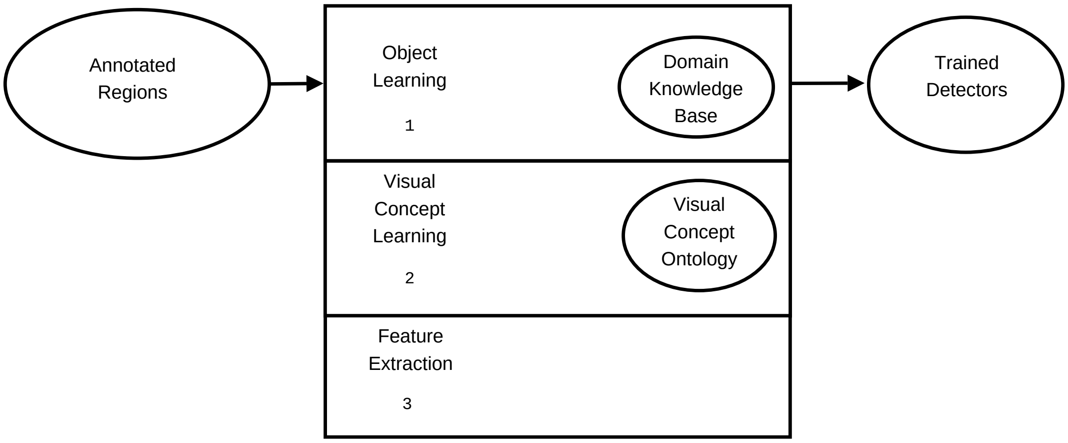


Figure 3.8: Object learning overview. The object learning process is based on visual concept learning. The input of this process is the set of annotated regions. The output is a set of visual concept detectors. The three levels of abstraction of the image interpretation problem structure the object learning process.

3.3.4 Object Categorization

Fig. 3.9 gives an overview of the proposed object categorization phase. This phase is based on three tasks:

1. Object categorization is initiated by categorization requests and produces categorization results. A categorization request is mainly composed of an image to interpret.A categorization result contains the object(s) recognized in this image coupled with a visual description in terms of visual concepts. Object categorization consists of a hierarchical exploration of the domain knowledge base. Each class of the knowledge base is used as an *hypothesis* which has to be *verified* in the image by visual concept detection. A hypothesis is a set of hypothetic visual concepts.

2. The role of the visual concept detection layer is to detect the hypothetic visual concepts in the image. The visual concept detection process requires automatic segmentation of the image and then feature extraction on the resulting regions.

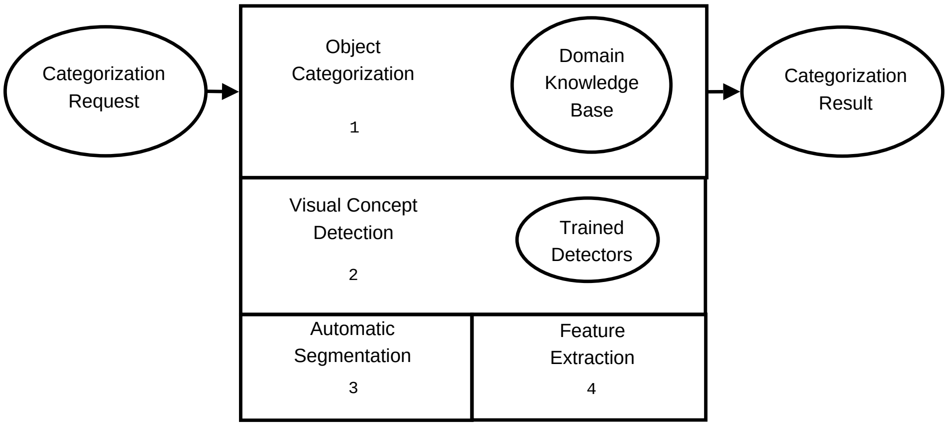


Figure 3.9: Object categorization phase overview. The *visual concept detection* layer stands as a mapping layer between high-level knowledge and low-level mechanisms (i.e. segmentation and feature extraction).

3. Segmentation produces regions which are used for feature extraction purposes. Segmentation is driven by the hypothetic visual description to perform region selection. (e.g. if the hypothetical object is *Big* then the small regions resulting from segmentation are ignored).

4. Feature extraction consists of transforming segmented regions of interest into numerical features (e.g. Gabor features for texture analysis).

3.4 Conclusion

Our work deals with the problem of object recognition and takes place in the framework of cognitive vision. Our goal is to propose an approach that has properties of re-usability and of convenience that enables an expert of a domain to build an operational object categorization system without having to cope with the low-level image processing problem and with the mapping problem.

The image interpretation problem can be decomposed in three sub-problems: *high-level interpretation*, *mapping* and *image processing*. The main contributions of this thesis are related to the problem of high-level interpretation and to the problem of mapping. These contributions rely on the combination of an ontology-driven knowledge acquisition process with machine learning techniques. Our approach is composed of three phases. A knowledge acquisition phase which reduces the semantic gap thanks to the visual concepts based description. The visual concepts are associated with low-level features which are directly computable at the image level. In other words, the visual concepts enable a user-friendly manipulation of image processing notions. A visual concept learning phase which consists of learning the mapping relations between low-level features and high-level knowledge by using the visual concepts as an intermediate layer. The third phase consists of using the results of learning and knowledge acquisition for object categorization purposes. This approach (i.e. the combination of machine learning techniques with ontological engineering) significantly reduces the knowledge acquisition bottleneck. The expert only has to provide a taxonomy/partonomy of the object classes described by visual concepts. This is a property of convenience. Moreover, the following chapters show that all the components of the proposed approach are generic. The visual concept ontology, the visual concept learning process and the object categorization process are not dedicated to a domain of interest. This means that the approach also has the property of re-usability.

Chapter 4

Knowledge Acquisition

4.1 Introduction

As explained in chapter 2, a cognitive vision system has to exhibit the faculties of *knowing* and *reasoning*. We propose an ontology-based methodology for developing these faculties with the goal of enabling the functionalities of *detection* and *recognition*. This chapter is related to one important aspect of cognitive vision: *representation*.

We are going to show how the domain knowledge base for semantic interpretation is acquired. Extracting domain knowledge means producing a hierarchical structure of domain object classes associated with their subparts (Fig. 4.1). This knowledge belongs to the domain of interest and is shared by the specialists of the domain (e.g. biologists, astronomers). It is important to note that domain knowledge is independent of any vision layer and can be reused for other purposes.

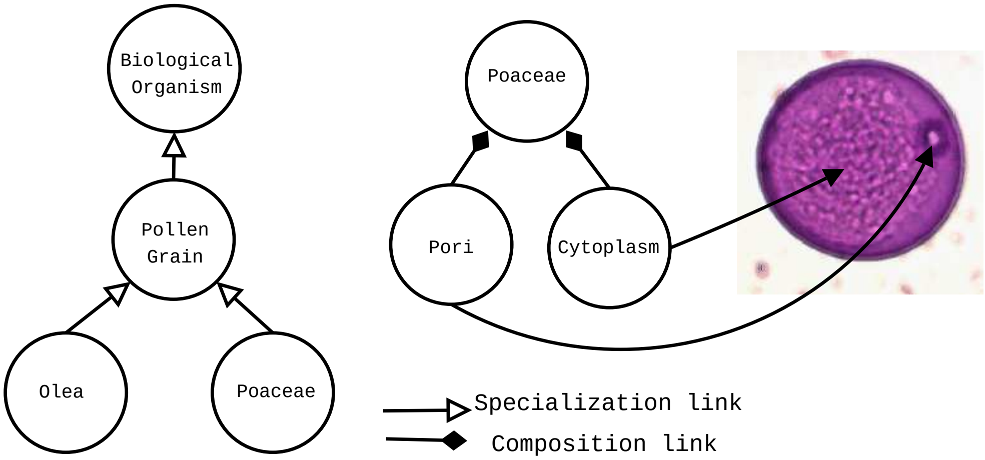


Figure 4.1: Domain knowledge structure. From the *left* to the *right*, a hierarchy of classes, a subpart tree, and an image representing the appearance of the Poaceae pollen grain in a light microscope.

We are also going to see how knowledge related to the problem of mapping image data to high-level concepts is acquired thanks to a visual concept ontology. This ontology can be considered as a guide which provides the vocabulary for the visual description of domain classes. During the visual concept ontology-driven description phase, the expert uses the vocabulary provided by the ontology to describe the objects of the domain. This task is performed in a user-friendly way with a graphical user interface described in section 4.5. As seen in fig. 4.2, ontological concepts are linked both to domain knowledge and to low-level vision numerical features. As a consequence, the visual concepts used for the visual description by the expert indicate which low-level features are useful to achieve object learning and recognition.

The overall knowledge acquisition process is depicted in fig. 4.3. The result of the description phase is a knowledge base composed of the visual concepts provided by the ontology associated with domain classes. For example, the visual concept *Circular Surface* provided by the ontology can be used to describe the geometry of a domain object.

This chapter is structured in five sections. Section 4.2 introduces a formalization of the entities involved in knowledge acquisition, learning and categorization. Section 4.3 provides a detailed presentation of the visual concept ontology involved in the knowledge acquisition process. Section 4.4 presents the low-level numerical features and image processing algorithms attached to each visual concept. Section 4.5 presents a knowledge acquisition tool. Finally, section 4.6 concludes the knowledge acquisition process.

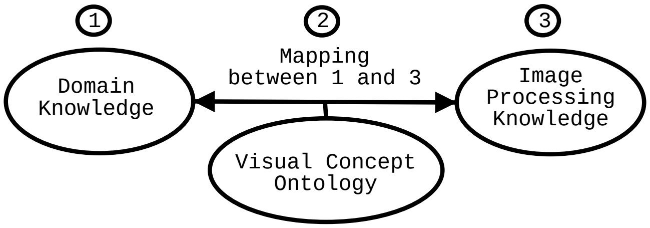


Figure 4.2: The visual concept ontology reduces the semantic gap between domain knowledge and image processing knowledge

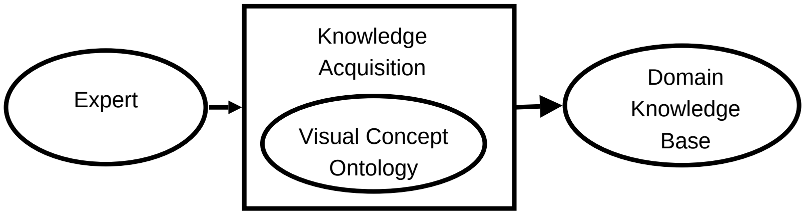


Figure 4.3: Knowledge acquisition process: the visual concept ontology guides the knowledge acquisition process.

4.2 Knowledge Formalization

In this section, we introduce a set of notations associated with the knowledge entities which are the support of knowledge acquisition, learning and categorization. Each visual concept is associated with low-level numerical features and algorithms.

Definition 1 *Let Θ be the set of all visual concepts. $\preceq_\Theta$ is a partial order between visual concepts. $\forall (C_i, C_j) \in \Theta^2, C_i \preceq_\Theta C_j$ means that C_i is a sub-concept of C_j.*

Definition 2 *Let $\mathcal{F}(C), C \in \Theta$ be the set of low-level features associated with C. $\forall (C_i, C_j) \in \Theta^2,\ C_i \preceq_\Theta C_j \Rightarrow \mathcal{F}(C_i) \subseteq \mathcal{F}(C_j)$.*

Definition 3 *Let Φ be the set of domain classes. $\preceq_\Phi$ is a partial order between domain classes (i.e. superclass attribute).*

Definition 4 *Let $\mathcal{A} \subset \Theta$ be the set of domain class visual attributes. $\mathcal{A}$ is a predefined subset of Θ. $\mathcal{A} = \{Geometry, Size$, Elongation, $Position$, Hue, $Brightness$, $Saturation$, Repartition, $Contrast$, $Pattern\}$. For a class $\alpha \in \Phi$, $\mathcal{A}_\alpha \subseteq \mathcal{A}$ is the set of visual attributes of α.*

Definition 5 *For $\alpha \in \Phi$, $\mathcal{S}_\alpha \subset \Phi$ is the set of subpart attributes of α.*

Definition 6 *For $\alpha \in \Phi$, $\mathcal{SR}_\alpha$ is the set of spatial relation attributes of α.*

Definition 7 *Let $a \in \mathcal{A}_\alpha$ be a visual attribute of $\alpha \in \Phi$. $\mathcal{V}_\alpha(a)$ is the set of possible values of a so that $\forall C \in \mathcal{V}_\alpha(a), C \preceq_\Theta a$ and $C \neq a$.*

Definition 8 *Let $s \in \mathcal{S}_\alpha$ be a subpart attribute of $\alpha \in \Phi$. $\mathcal{V}_\alpha(s)$ is the set of possible values of s.*

Definition 9 *Let $sr \in \mathcal{SR}_\alpha$ be a spatial relation attribute of $\alpha \in \Phi$. $\mathcal{V}_\alpha(sr)$ is the set of possible values of sr.*

These definitions are important and are used in chapter 5 and also in chapter 6.

The most important knowledge representation formalisms have been presented in chapter 2. Commonly used techniques are formal logics, fuzzy logics, frames, semantic nets or description logics (DL). We want an expressive yet user-friendly formalism. The two main candidates are: frames and description logics. As explained in [Neumann and Weiss, 2003], state-of-art DL reasoners do not offer all the reasoning services required for image interpretation (e.g. part-whole reasoning). Moreover, the user-friendliness of frame-based formalisms enables easy modification and extension of knowledge bases.

We use a frame based formalism (which is well adapted to the representation of taxonomical expert knowledge). An example of visual description of a pollen grain Poaceae (fig. 4.1) is given in table 4.1. This description is the way experts describe this pollen grain viewed by using a light microscope with a magnification ×60. The color description depends on the type of *dye* used. As seen in table 4.1 ($\alpha = Poaceae$ and $\mathcal{A}_\alpha = \{Geometry,\ Size,\ Hue,\ Brightness,\ Pattern,\ Contrast\}$) , a domain class is described through five categories of attributes :

1. *Subpart attributes.* In table 4.1, $\mathcal{S}_\alpha = \{pori\}$ and $\mathcal{V}_\alpha(pori) = \{PoriWithAnulus\}$.

2. *Spatial attributes : Geometry, Size, Elongation, Position.* In table 4.1, $\mathcal{V}_\alpha(Geometry) = \{CircularSurface,\ EllipticalSurface\}$, $\mathcal{V}_\alpha(Size) = \{LargeSize,\ AverageSize,\ SmallSize\}$.

3. *Color attributes : Hue, Brightness, Saturation.* In table 4.1, $\mathcal{V}_\alpha(Hue) = \{Pink\}$ and $\mathcal{V}_\alpha(Brightness) = \{Dark\}$.

4. *Texture attributes : Repartition, Contrast, Pattern.* In table 4.1, $\mathcal{V}_\alpha(Pattern) = \{GranulatedTexture\}$ and $\mathcal{V}_\alpha(Contrast) = \{Slight\}$.

5. *Spatial relation attributes.* In table 4.1, $\mathcal{SR}_\alpha = \{r1\}$ and $\mathcal{V}_\alpha(r1) = \{NTTP(poaceae, pori),\ TTP(Poaceae, pori)\}$.

Any type of attribute can be *weighted* depending on its importance. For a class α, the weight associated with an attribute a is noted $w_{\alpha,a} \in [0, 1]$. The default weight value is 1.

4.3 A Visual Concept Ontology

In this section, we propose a visual concept ontology. This ontology can be considered as a guide which provides a vocabulary for the visual description of domain classes. It is important to note that the proposed ontology is not application-dependent and should be considered as an extensible basis. We have structured this ontology in three main parts. The first one contains *texture concepts*, the second one contains *color concepts* and the last one is made of *spatial concepts.* Each part of this ontology is detailed in the next subsections.

4.3.1 Texture Concepts

This part of the ontology has been inspired by results from the cognitive science community.

The experiment conducted in [Rao and Lohse, 1993] and [Bhushan et al., 1997] identifies three main dimensions in the texture perception cognitive process. A subset of 56 Brodatz texture images has been given to 20 persons who were asked to classify them in different clusters. The clusters were formed by evaluating the following symbols (between 1 and 9): contrast, repetitiveness, granularity, randomness, roughness, density, directionality, complexity, coarseness, regularity, orientation. The clusters were obtained by applying hierarchical clustering and multi-dimensional scaling techniques. Some samples of the texture concepts used this experiment can be found in table 4.2.

Class	*Poaceae*
{	
SuperClass:	*PollenWithPori*
SubParts:	
pori	[*PoriWithAnulus*]
SpatialAttributes :	
Geometry :	[**CircularSurface EllipticalSurface**]
Size :	[**LargeSize AverageSize SmallSize**]
ColorAttributes :	
Hue:	[**Pink**]
Brightness:	[**Dark**]
TextureAttributes :	
Pattern:	[**GranulatedTexture**]
Contrast:	[**Slight**]
SpatialRelations :	
r1:	[**NTTP**(*Poaceae,pori*) **TTP**(*Poaceae,pori*)]
}	

Table 4.1: High level description of domain class Poaceae. Visual concepts provided by the ontology are in **bold face**. Attribute names are in SMALL CAPS. Knowledge provided by the expert is in *italic*.

Each perceptual dimension constitutes an important element in texture perception. Each perceptual dimension is seen as an abstraction of a set of texture visual concepts (Fig. 4.4). From this study, we have built an ontology of texture concepts. Three quantifiers (i.e. *Not*, *Slightly*, *Strongly*) are also integrated in this ontology and can be used to give a finer texture description. Examples of such quantifications are: *Not Regular* or *Strongly Directional*.

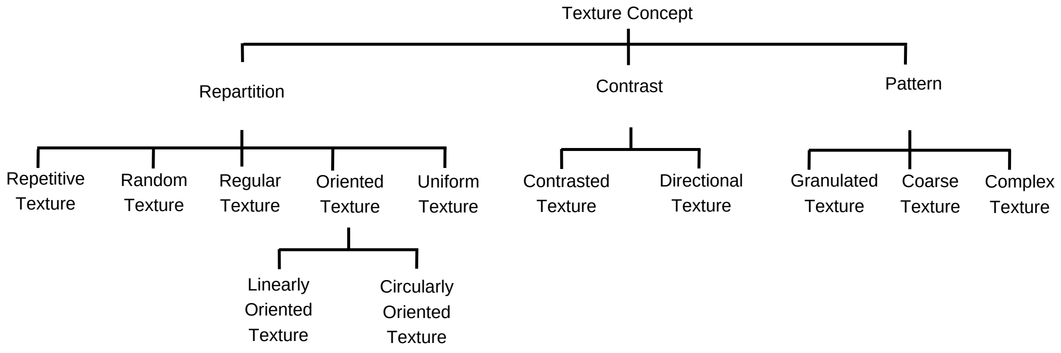


Figure 4.4: Texture concept hierarchy

4.3.2 Color Concepts

This part of the ontology is derived from the ISCC-NBS (Inter-Society Color Council-National Bureau of Standards) color dictionnary. An interesting reflexion on the validity of this dictionnary is given in [Miller and Johnson-Laird, 1976]. Three kinds of notions are included: hue, brightness and saturation concepts. There are 28 hue concepts (Table 4.3) which can be combined with five brightness concepts (*Very Dark, Dark, Medium, Light, Very Light*) and four saturation concepts (*Grayish, Moderate, Strong, Vivid*). Certain combinations of brightness and saturation concepts have a perceptual meaning. For instance, the concept *Brillant* is an association of the *Light* and *Strong* concepts. Axioms are contained in the ontology so as to express those kinds of associations.

4.3.3 Spatial Concepts

This part of the ontology is used for describing domain objects from a spatial point of view. There are four kinds of spatial concepts in the ontology: 9 position concepts, 21 geometric concepts, 3 elongation concepts, and 3 size concepts. The hierarchy of geometric concepts is shown in fig. 4.6. A formalization of a similar approach based on a combination of geometric shapes can be found in [Sciascio et al., 2002]. The size of an object can also be described by the following concepts: *LargeSize*, *AverageSize*, *SmallSize*. Elongation concepts are the following: *LargeElongation*, *AverageElongation*, *SmallElongation*. Position concepts are shown in fig. 4.5.

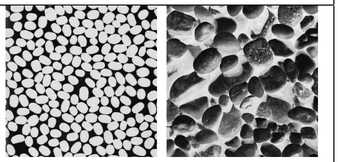
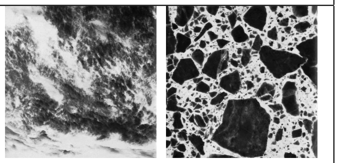
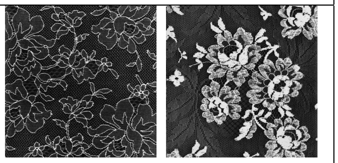
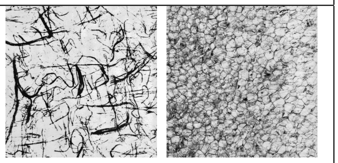
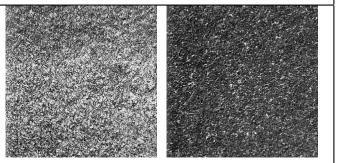
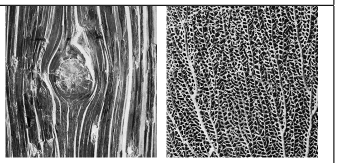
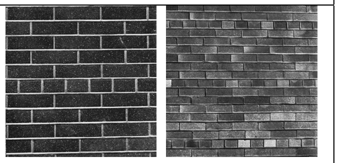
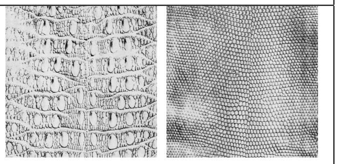
Cluster	Visual Concept(s)	Texture Samples
A	Granulated	
B	Random, Not Granulated, Not Repetitive	
C	Not Random, Not Repetitive, Not Directional	
D	Random, Repetitive	
E	Random	
F	Directional	
G	Repetitive, Oriented, Uniform	
H	Directional	

Table 4.2: Clusters obtained by an experiment involving 20 persons. The second columns contains the visual concepts associated with the clusters. The last column gives samples coming from the Brodatz texture set [Brodatz, 1966].

Red	Purple
Reddish Orange	Reddish Purple
Orange	Purplish Red
Orange Yellow	Purplish Pink
Yellow	Pink
Greenish Yellow	Yellowish Pink
Yellow Green	Brownish Pink
Yellowish Green	Brownish Orange
Green	Reddish Brown
Bluish Green	Brown
Greenish Blue	Yellowish Brown
Blue	Olive Brown
Purplish Blue	Olive
Violet	Olive Green

Table 4.3: Set of hue concepts

Top Left	Top	Top Right
Center Left	Center	Center Right
Bottom Left	Bottom	Bottom Right

Figure 4.5: The set of 9 position concepts contained in the visual concept ontology.

4.3.4 Spatial Relations

Frame of Reference

The notion of spatial relations is dependent on the notion of frame of reference. The concept of frame of reference is also found in the ontology. Two kinds of frames of reference are considered in the visual concept ontology:

1. The *image frame of reference* (also called the *egocentric frame of reference*). This frame of reference is used to describe the position of an object in the image but also distance and orientation relations between several objects in the image.

2. The *intrinsic frame of reference*. This frame of reference is used for describing the orientation relations and the distance relations between the subparts of an object of interest. This frame of reference is relative to a given object of interest.

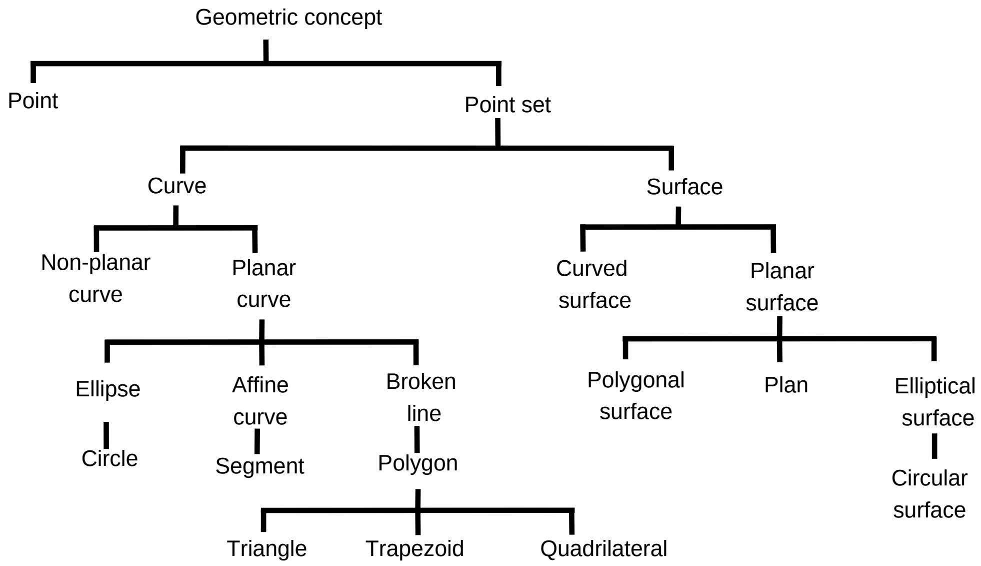


Figure 4.6: Geometric concept hierarchy. The visual concept ontology only contains 2-D geometric concepts.

Topological Relations

We have also added a set of 8 spatial relations based on the RCC-8 model that can be used to define relations between objects and their subparts. An overview of *qualitative spatial reasoning* can be found in [Cohn and Hazarika, 2001]. These relations are enumerated in Table 4.4 and graphically represented in fig. 4.7.

RCC-8 relation	Meaning
DC(X,Y)	X disconnected from Y
EC(X,Y)	X externally connected to Y
EQ(X,Y)	X equals Y
PO(X,Y)	X partially overlapping Y
TPP(X,Y)	X tangential proper part of Y
TPP-1(X,Y)	X has tangential proper part Y
NTPP(X,Y)	X nontangential proper part of Y
NTPP-1(X,Y)	X has nontangential proper part Y

Table 4.4: RCC-8 relations and their meaning

By using the results presented in [Clementini et al., 1993], the generic RCC-8 relations can be specialized for taking into account spatial relations between the following entities: lines, points and surface. The authors of this work propose a set of spatial relations designed for user-friendly interaction. A graphical representation of the most important Curve/Curve and Curve/Surface relations is presented in tables 4.5 and 4.6.

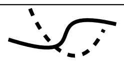
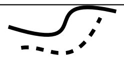
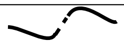
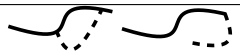
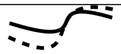
Curve/Curve Relation	Graphical Representation
Cross	
Disjoint	
In	
Touch	
Overlap	

Table 4.5: The 11 Curve/Curve relations contained in the ontology and their graphical representation

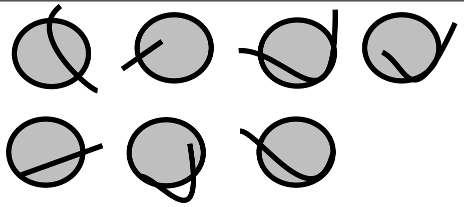
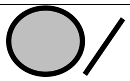
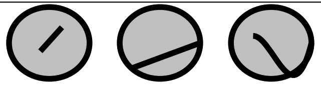
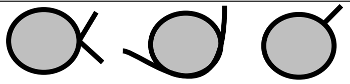
Curve/Surface Relation	Graphical Representation
Cross	
Disjoint	
In	
Touch	

Table 4.6: The 18 Curve/Surface relations contained in the ontology and their graphical representation

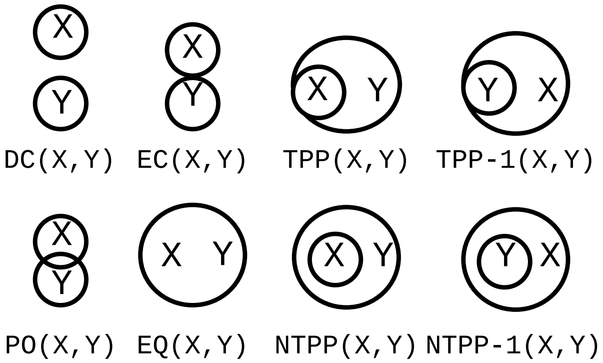


Figure 4.7: RCC-8 graphical representation

Orientation Relations

Four concepts enable orientation relations description: *RightOf*, *LeftOf*, *Above*, *Below*. Note that for orientation relations between an object and its subparts, the intrinsic frame of reference of the main object is considered. For orientation relations between several objects in the image, the image frame of reference is considered.

Distance Relations

Two concepts enable distance relations description: *Far*, *Close*. These concepts can be combined with the quantifiers *Very*, *Slightly* and *Not*. Examples of quantifications are the following: *Not Far*, *Very Close*.

4.3.5 Acquisition Context

Experts often observe the objects of their domain in precise observation conditions. For example, when using a microscope, magnification or lighting conditions are controlled. Providing contextual information is absolutely necessary. As shown in Fig. 4.8, context information is the link between domain knowledge and the appearance of the objects of interest. Context conditions the resulting acquired images. This implies a relation between the visual description of the objects of interest and the context of acquisition. Acquisition context depends on the application domain. That is why the acquisition context hierarchy given in Fig. 4.9 can be extended and adapted for a particular domain.

4.4 Link with the Low-Level Vision Layer

The previous subsections have introduced the structure of the proposed visual concept ontology. Any knowledge base resulting from the ontology-driven knowledge acquisition process is for classification and learning purposes. During the classification or the learning of visual concepts, numerical features are computed. To be interpreted as visual concepts,

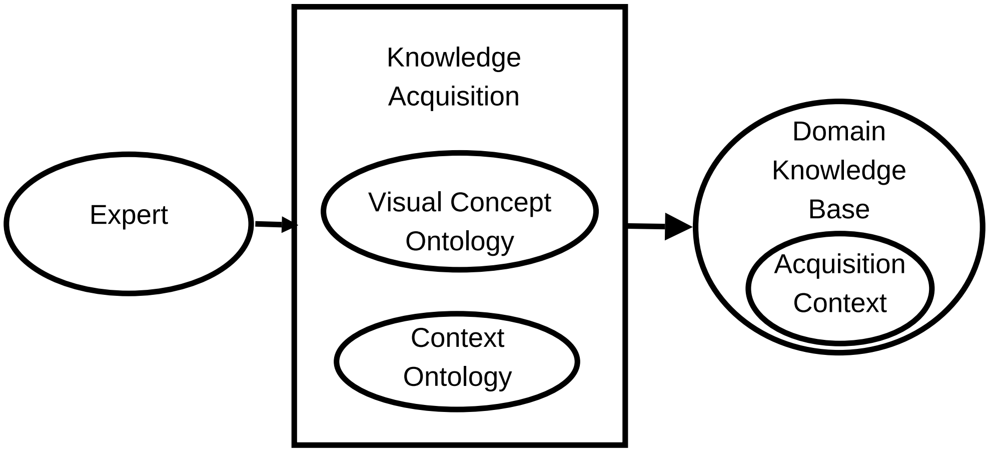


Figure 4.8: Contextual knowledge in the global knowledge acquisition process. The visual description of the objects of interest is valid only in a given context. The visual description may remain stable for slight changes of context. For completely different contexts, visual description has to be changed.

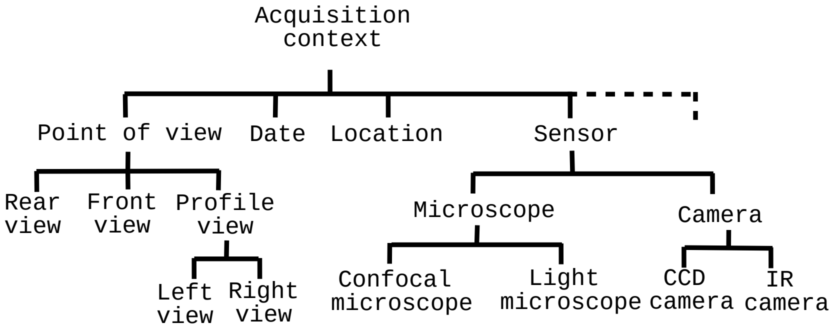


Figure 4.9: Acquisition context concept hierarchy

a link must be established between computed numerical features and symbolic visual concepts.

This section is structured in three parts. The first part presents low-level features associated with spatial concepts (e.g. $\mathcal{F}(Geometry)$, $\mathcal{F}(Elongation)$ and $\mathcal{F}(Size)$). The second part present low-level features associated with color concepts (e.g. $\mathcal{F}(Hue)$). The third part present low-level features associated with texture concepts (e.g. $\mathcal{F}(Pattern)$). The value of these features can be computed on image data such as a Region by feature extraction algorithms.

4.4.1 Spatial Features

The link between symbolic visual concepts and numerical features is manually defined. A set of calculated shape features is given in Table 4.7. Subsets of these features are associated with other visual concepts. For example, the ratio $Length/Height$ computed for a region of interest is used to characterize the *Elongation* visual concept. The features *Area* and *Perimeter* are associated with the visual concept *Size*.

Sift Features

Scale-invariant feature transform (Sift) features have been introduced in [Lowe, 1999]. These features belong to the class of local image features. The are well adapted for characterizing small details. They are invariant to image *scaling*, image *translation*, and partially invariant to *illumination changes* and *affine* for 3D projection. First, features are detected through a staged filtering approach that identifies stable points in scale space. The result of this detection, is a set of key local regions. Then, given a stable location, scale, and orientation for each key point, it is possible to describe the local image regions in a manner invariant to these transformations.

Key locations are selected at maxima and minima of a difference of Gaussians applied in scale space. The input image I is first convolved with the Gaussians function to give an image A. This is then repeated a second time with a further incremental smoothing to give a new image B. The difference of Gaussians function is obtained by subtracting image B from A. This difference of Gaussians is formally expressed as:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \tag{4.1}$$

with k corresponding to the strength of smoothing and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp -(x^2 + y^2)/2\sigma^2 \tag{4.2}$$

This differentiation process is repeated with different values of k.

A change of scale consists of sampling the smoothed images by using a bilinear interpolation. The combination of scaling and smoothing produces a scale space pyramid. An overview of the scale/space construction is shown in fig. 4.10.

Minima and extrema detection of $D(x, y, \sigma)$ uses this scale space pyramid and is achieved by comparing each sample point to its neighbors in the current image and 9 neighbors in the scale above and below. It is selected only if it is larger than all its neighbors or smaller than all its neighbors.

The result of this selection is a set of key-points which are assigned a location, a scale and an orientation (i.e. obtained by gradient orientation computation).

The last step consists of assigning a numerical vector to each keypoint. The 16 × 16 neighborhood around the key location is divided into 16 sub-regions. Each sub-region is used to compute an orientation histogram. Each bin of a given histogram corresponds to the sum of the gradient magnitude of the pixels in the sub-region. The final numerical vector f associated with a keypoint is of dimension 128. For an image I, a set of such numerical vectors is computed.

Descriptor	**Formula**
Length (L)	Maximum projection
Width (W)	Maximum orthogonal to length
$Ratio\ \frac{L}{W}$	$\frac{Width}{Length}$
Area (A)	Number of pixels
Form Factor	$\frac{4\pi A}{P^2}$
Perimeter	Perimeter length
Roundness	$\frac{4A}{\pi L^2}$
Equivalent circular diameter (ECD)	$\sqrt{\frac{4A}{\pi}}$
Compactness	$\frac{ECD}{L}$
Box area (BXA)	Bounding rectangle
Box ratio	$\frac{A}{BXA}$
Convex hull area (CHA)	Area of convex hull
Convex hull perimeter (CHP)	Perimeter of convex hull
Solidity (S)	$\frac{A}{CHA}$
Concavity (CCav)	$CHA - A$
Convexity (CVex)	$\frac{CHP}{P}$

Table 4.7: Examples of numerical features associated with geometric visual concept. These features are well-adapted for characterizing the geometry of a region.

We use Sift features by clusters of key-points as explained in [Csurka et al., 2004]. This bag of keypoints method is based on vector quantization of the Sift features extracted in the image.

The main steps of this method are:

1. Detection and description of image patches.

2. Assigning patch descriptors to a set of predetermined clusters (a vocabulary) with a vector quantization algorithm.

3. Constructing a bag of keypoints, which counts the number of patches assigned to each cluster.

4. Applying a multi-class classifier, treating the bag of keypoints as the feature vector, and thus determine which category or categories to assign to the image.

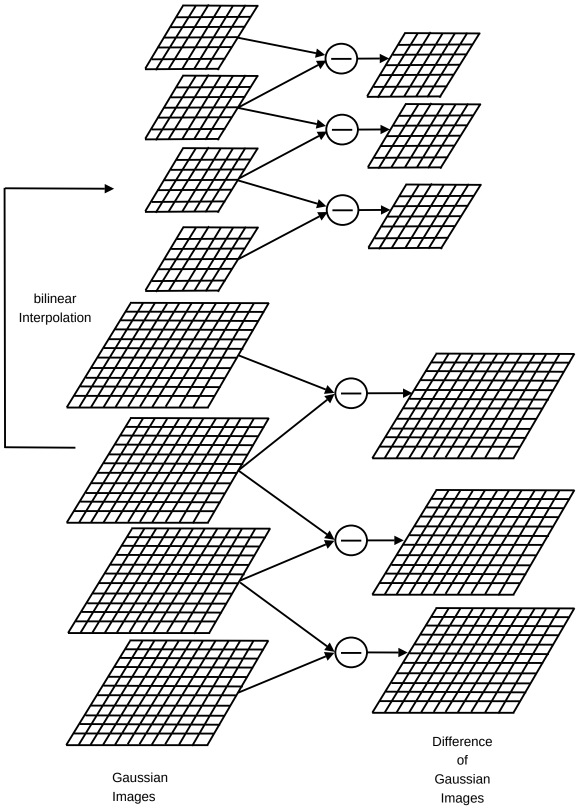


Figure 4.10: Construction of the scale/space pyramid. Gaussian images are used to obtain difference of Gaussian images. Scale changes are obtained by bilinear interpolation of each Gaussian images. Minima and extrema detection is performed on this pyramid.

4.4.2 Color Features

Color histograms

Considering a three-dimensional color space (x, y, z), quantized on each component to a finite set of colors which correspond to the number of bins N_x, N_y, N_z, the color of the image I is the joint probability of the intensities of the three color channels. Let $i \in [1, N_x],\ j \in [1, N_y]\ and\ k \in [1, N_z]$. Then, $h(i, j, k) = Card\{p \in I \mid color(p) = (i, j, k)\}$. The color histogram H of image I is then defined as the vector $H(I) = (..., h(i, j, k), ...)$.

Color Coherence Vectors

Color coherence vectors have been introduced in [Pass et al., 1996]. A color coherence vector can be seen as a color histogram where pixels in each bin are split between coherent and non coherent pixels. A pixel is said to be coherent if it belongs to a large group of pixels of the same color. Let $i \in [1, N_x],\ j \in [1, N_y]\ and\ k \in [1, N_z]$: $\forall 1 \leq i \leq N_x$, $\forall 1 \leq j \leq N_y$, $\forall 1 \leq k \leq N_z$:

$$\alpha(i, j, k) = Card\{p \in I \mid color(p) = (i, j, k) \wedge Coh(p) = 1\} \tag{4.3}$$

$$\beta(i, j, k) = Card\{p \in I \mid color(p) = (i, j, k) \wedge Coh(p) = 0\} \tag{4.4}$$

The color coherence vector CCV of image I is then defined as the vector $CCV(I) = (..., (\alpha(i, j, k), \beta(i, j, k)), ...)$, where $\alpha(i, j, k)$ and $\beta(i, j, k)$ are respectively the coherent and incoherent number of pixels of color (i, j, k).

4.4.3 Texture Features

Grey-Level Cooccurence Matrices (GLMC)

This texture feature is based on the grey level cooccurence or two-dimensional spatial dependency of the grey levels for a fixed distance and/or angular spatial relationship. Let $D = \{(d_{x_i}, d_{y_i})\}$, a set of displacement vectors. For an image I coded on N grey-levels, for two grey-levels $g_1 \in [1, N]$ and $g_2 \in [1, N]$, and for a fixed value of i, the co-occurence matrix C_D (which is of dimension $N \times N$) is defined as:

$$\begin{aligned} C_D(g_1, g_2) = Card\{((x, y), (x', y') \mid I(x, y) = g_1\ \wedge\ I(x', y') = g_2 \\ \wedge\ x = x' + d_{x_i}\ \wedge\ y = y' + d_{y_i}\} \end{aligned} \tag{4.5}$$

The resulting GLMC can be normalized and then, $C_D(g_1, g_2)$ corresponds to the probability that two pixels are at a distance corresponding to the norm of the displacement vector and have the grey-values (g_1, g_2).

The following statistics originally proposed in [Haralick, 1979] are then computed from C_D:

1. The maximum element of C_D: $\max_{ij}\{C_D(i,j)\}$.
2. The element difference of order k: $\sum_i \sum_j C_D(i,j)(i-j)^k$.
3. The element inverse difference of order k: $\sum_i \sum_j C_D(i,j)/(i-j)^k$.
4. Entropy: $-\sum_i \sum_j C_D(i,j)\ log\ C_D(i,j)$.
5. Uniformity: $\sum_i \sum_j C_D(i,j)^2$.

Gabor Features

We use the Gabor Wavelet Transforms as suggested in [Manjunath and Ma, 1996]. In addition to good performances in texture discrimination and segmentation, the justification for Gabor filters is also supported through psychophysical experiments. Texture analyzers implemented using 2-D Gabor functions produce a strong correlation with actual human segmentation [Reed and Wechsler, 1990]. Gabor functions are Gaussians modulated by complex sinusoids. In two dimensions they take the form:

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}) + 2\pi jWx\right) \tag{4.6}$$

A dictionary of filters can be obtained by appropriate dilatations and rotations of $g(x,y)$ through the generating function:

$$g_{mn}(x,y) = a^{-m}g(x',y'),\ m = 0,1,...,S-1 \tag{4.7}$$

$$x' = a^{-m}(xcos\ \theta + ysin\ \theta),\ y' = (-xsin\ \theta + ycos\ \theta) \tag{4.8}$$

where $\theta = n\pi/K$, K the number of orientations, S the number scales in the multiresolution, and $a = (U_h/U_l)^{-1/S-1}$ with U_l and U_h the lower and upper center frequencies of interest. A compact representation needs to be derived for learning and classification purposes. Given an image $I(x,y)$, its Gabor wavelet transform is then defined as:

$$W_{mn}(x,y) = \int I(x,y)g_{mn} * (x-x_1, y-y_1)\, dx_1 dy_1 \tag{4.9}$$

where $*$ represents the complex conjugate. The mean μ_{mn} and the standard deviation σ_{mn} of the magnitude of the transform coefficients are used to represent the image.

$$\mu_{mn} = \int\int |W_{mn}(x,y)|\, dx\, dy,\ and\ \sigma_{mn} = \sqrt{\int\int (|W_{mn}(x,y)| - \mu_{mn})^2\, dx\, dy} \tag{4.10}$$

A feature vector is then constructed using μ_{mn} and σ_{mn} as feature components:

$$\mathbf{f} = [\mu_{00}\ \sigma_{00}\ \mu_{01}\ \sigma_{01}\ ...\ \mu_{mn}\ \sigma_{mn}] \tag{4.11}$$

4.4.4 Conclusion

Each visual concept of the ontology is associated with low-level features and with image processing algorithm able to compute these features. This gives a *procedural semantics* to the visual concept ontology. We have chosen a great variety of low-level features which enable the characterization of an image (or a region) from both the photometric and the geometric point of view. We have also introduced the notion of key-point which are very useful to characterize small elements in an image which are difficult to isolate by classical region segmentation methods. All these techniques are widely used in the computer vision community and have proved their efficiency.

4.5 A Knowledge Acquisition Tool for Visual Description

4.5.1 Overview

Section 4.3 contains details about the structure of a visual concept ontology. To be used as a guide for the description of domain objects, a dedicated graphical tool called OntoVis has been developed. This tool is currently able to carry out two distinct tasks:

1. domain knowledge definition

2. visual concept ontology-driven symbolic description of concepts and their subparts

The output result of the acquisition process is a knowledge base composed of domain classes described by visual concepts provided by the ontology. The Java programming language has been used to create this tool. An overview of the tool can seen in fig. 4.11.

4.5.2 Tool Characteristics

Ontovis enables domain knowledge acquisition. As can be seen in Fig. 4.12 and Fig. 4.13, domain knowledge is organized as a taxonomy of domain classes in a specialization tree. This approach is natural for people who are familiar with a taxonomic approach (e.g. biologists). Whenever a class is added to the tree, the visual concept ontology is displayed on the screen. The user is then able to describe a new class with the terminology contained in the ontology. As previously explained, a class can be composed of subparts (*subparts* attribute).

Subpart description is performed in the same way as the description of domain classes. Note that the subpart tree is a composition tree and not a specialization tree. Every domain class has an associated subpart tree (see Fig. 4.12).

4.6 Conclusion

The notion of visual concept ontology has been introduced. Its structure is based on three distinct notions: texture, color and spatial concepts. The notion of context ontology has

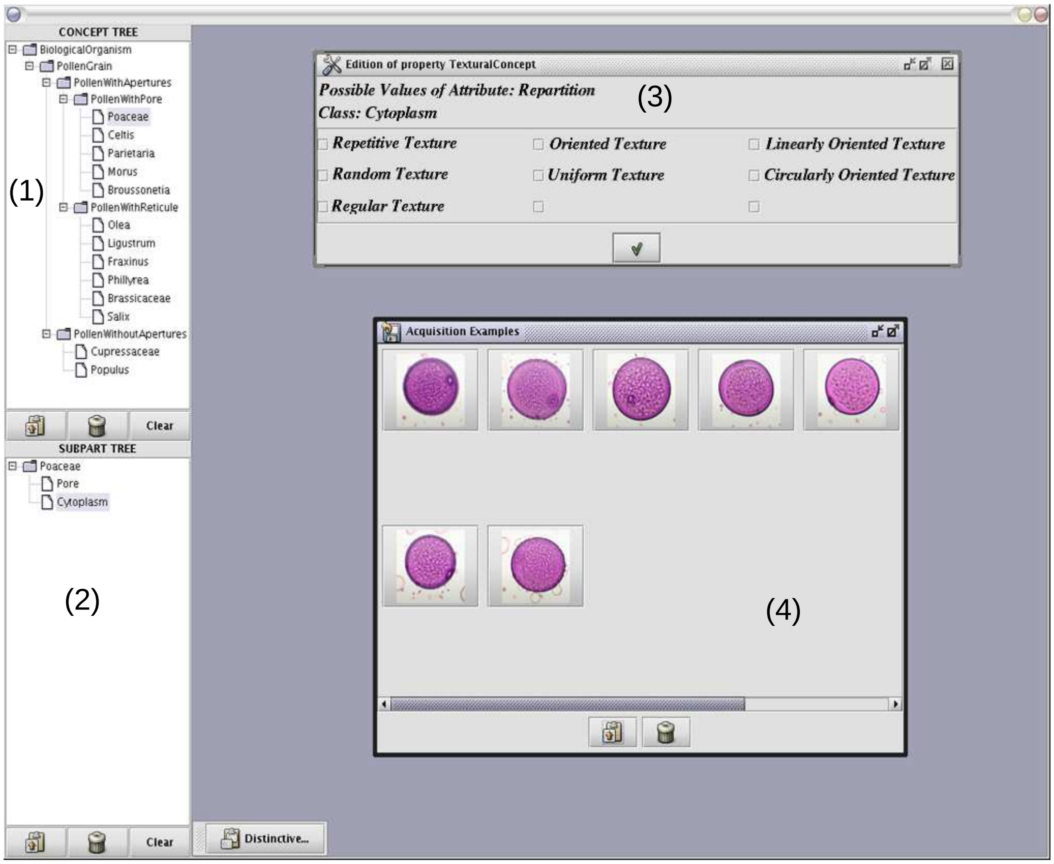


Figure 4.11: Overview of Ontovis. This screenshot results from a knowledge acquisition session in the domain of palynology. This tool is composed of four main parts. (1) the domain class hierarchy. (2) The subparts associated with each domain class. (3) Visual concepts used for the visual description. (4) Image samples of the objects of interest used during the learning phase.

also been introduced. This ontology can be used as a guide for describing the objects from a specific domain. A set of numerical features is associated with each category of visual concepts.

We propose a reusable and extensible methodology for acquiring knowledge related to complex object visual description. One important aspect of the knowledge acquisition process is that it is designed to acquire the appearance of the objects of interest in a given context. This means that a change of context implies modifications of acquired knowledge.

The concrete implementation of this methodology currently involves the following elements :

- A visual concept ontology composed of 16 texture concepts, 37 color concepts, 28 spatial concepts, 16 context concepts, 4 quantifiers and 43 spatial relations. The total number of concepts is 144.
- Six feature extraction algorithms used for the characterization of color, texture, and

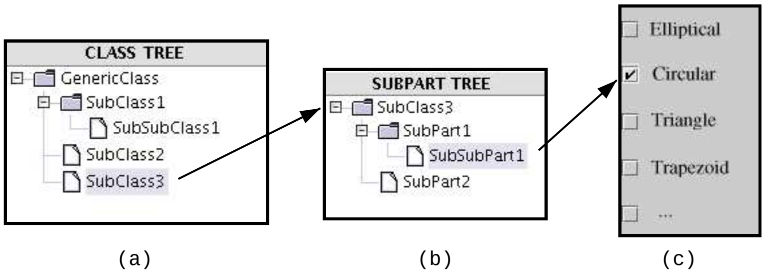


Figure 4.12: Description of subpart ”SubSubPart1” of class ”SubClass3”. (a) Domain classes hierarchy (b) Subpart Tree associated with SubClass3 (c) Visual concepts proposed for the description

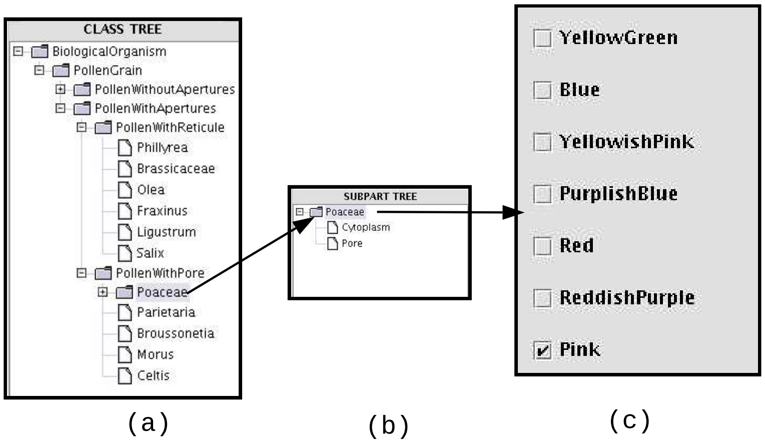


Figure 4.13: Description of domain class Poaceae. (a) Domain classes hierarchy (b) Subpart tree associated with domain class Poaceae (c) Hue visual concepts used for the description

spatial visual concepts. For instance, Gabor features and co-occurence matrices are associated with texture visual concepts. Another example is the use of color coherence vectors which are associated with color concepts. All these features have been widely used in the literature and have proved their efficiency.

- A knowledge acquisition tool which enables the acquisition of domain knowledge as a taxonomy/partonomy of domain classes described by visual concepts.

Some recent research efforts dedicated to ontologies which have common concepts with the proposed visual concept ontology can be found in the literature. In [Coenen and Visser, 1999], a generic ontology for spatial reasoning is proposed. This work proposes an interesting formalized framework which has been used for spatial reasoning in Geographic Information Systems (GIS). It is rather focused on the logical fundations of spatial reasoning. The proposed ontology is not linked to image processing algorithms or features. In [Mezaris et al., 2004], an *object ontology* is proposed. This ontology contains a set of color concepts and of spatial concepts. Texture concepts and spatial relations are not considered in this work.

The methodology proposed in this chapter is intended to establish a link between the semantic visual description provided by the experts and the low-level numerical features useful for object learning and recognition. This link leans on a visual concept ontology composed of visual concepts associated with numerical features. The visual concept ontology hides the complexity of image processing algorithms and features and enables any domain expert to manipulate image processing notions. This brings the property of *convenience* to our approach. The proposed visual concept ontology is generic and is thus important from the point of view of the *re-usability* of our approach.

The next chapter shows how this knowledge is used as the support for visual concept learning.

Chapter 5

Visual Concept Learning

5.1 Introduction

Chapter 4 has described a knowledge acquisition process that leads to a knowledge base structured as a taxonomy/partonomy of domain classes described by visual concepts. This visual description is valid for a given context. The knowledge acquisition phase is useful to capture expert knowledge. This is particularly important in application domains where the number of experts is decreasing. Another important aspect of the knowledge acquisition phase is that the semantic gap between domain expertise and low-level features and algorithms is reduced thanks to the visual concepts. Indeed, Each visual concept C is associated with a set of low-level features $\mathcal{F}(C)$. This association indicates which low-level features are useful to characterize an image region in terms of the values of domain class attributes used during knowledge acquisition. For a domain class $\alpha \in \Phi$ the set of *possible* values of its attribute a is defined as $\mathcal{V}_\alpha(a), a \in \mathcal{A}_\alpha$. For instance, for $a = Size$, $\mathcal{V}_\alpha(a) = \{LargeSize,\ SmallSize\}$, and $\mathcal{F}(Size) = \{area,\ perimeter\}$, the low-level features *area* and *perimeter* are meaningful to compute the size of a region of interest in terms of different values such as *LargeSize* or *SmallSize*.

The goal of visual concept learning is to ground the possible values of domain class attributes by producing a set of *visual concept detectors*. The role of a visual concept detector is, for an image region R and visual concept C, to compute the confidence value associated with the hypothesis "R is a representative sample of C". C is a value of one domain class attribute. The computed confidence value is done by feature extraction (i.e. computing the values of the features $\mathcal{F}(C)$). An example of a texture visual concept detection is given in fig 5.1.

Machine learning techniques are used for coping with this problem. As seen in fig. 5.2, annotated samples provided by the expert of the application domain are involved in the learning process.

This chapter is structured in three main sections. **Section 5.2** presents three different approaches for obtaining region samples annotated with visual concepts. The first approach requires a manual segmentation of each image sample of the image training set.

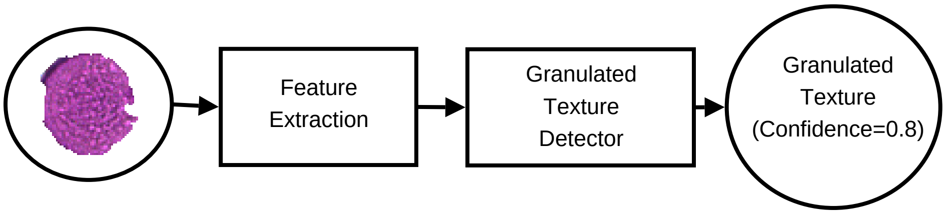


Figure 5.1: From an image region to a confidence degree associated with a visual concept. In this case, the region is recognized as being *Granulated* with a confidence value of 0.8. *Granulated Texture* is a possible value of the attribute *Pattern*. The features extracted are defined by $\mathcal{F}(Pattern)$ (i.e. Gabor features and co-occurence matrices).

This manual segmentation process is based on a tool called intelligent scissors. This tool does not require selecting manually each pixel of a boundary. We also show how this approach can be combined with a contour propagation algorithm based on B-spline snakes for enabling easier segmentation of spatio-temporal sequences.

The second approach is based on three-dimensional models of the objects of interest. Regions are obtained by projection of these models. This is a very convenient approach for obtaining images from different point of views of the objects of interest. The third approach is based on unsupervised machine learning techniques. All the images of the image training set are first automatically segmented. The resulting regions are then clustered by a k-means algorithm in order to obtain clusters of similar regions. The notion of similarity is based on each category of visual concept (e.g. size, hue, geometry).

Section 5.3 shows how the annotated image samples are used to obtain a set of visual concept detectors by feature extraction, feature selection and finally by training Support Vector Machines (SVM).

Section 5.4 is dedicated to the presentation of an algorithm which uses the domain knowledge base acquired during knowledge acquisition in order to achieve object learning.

5.2 Image Samples Segmentation and Annotation

In order to achieve symbol grounding, samples of the visual concepts used during the knowledge acquisition phase have to be provided. This section presents three different approaches designed to obtain image regions annotated by visual concepts. This set of annotated regions is defined as $\mathcal{AR} = \{(R_i, C_i)\}$. Each element of $\mathcal{AR}$ is a region R_i annotated by a visual concept C_i. We propose three approaches for building this set of annotated regions: a manual approach, a three-dimensional model based approach, and a weakly-supervised approach. The following sections are dedicated to the presentation of these approaches.

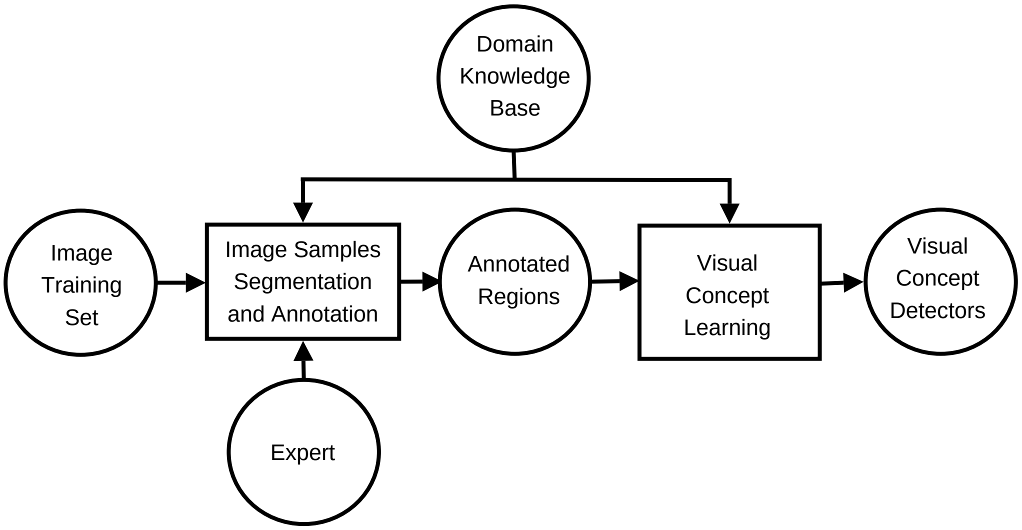


Figure 5.2: Overview of the visual concept learning process. The input of the visual concept learning process is an image training set. Each image of this set has to be segmented into regions annotated by one or several visual concepts used during knowledge acquisition. The visual concept learning process computes a set of visual concept detectors from the annotated image regions.

5.2.1 Manual Approach

Providing annotated image regions is done manually in three steps: by choosing a set of training images, then by manual segmentation of these images, and finally by annotating the resulting regions by visual concepts.

First of all, a set of image samples has to be chosen. As seen in fig. 5.3, a specific module of OntoVis is dedicated to image samples management. The expert can add or remove image samples.

Then, a region of interest is selected with an interactive drawing tool called *intelligent scissors*. This technique is presented in details in [Mortensen and Barrett, 1998] and has been improved in [Mortensen and Barrett, 1999]. Intelligent scissors allow objects within images to be extracted quickly and accurately using simple gesture motions with a mouse. When the gestured mouse position comes in proximity to the edge of an object, the manually drawn contour is wrapped around the object. The problem of contour detection is posed as an optimal path search in a weighted graph. Each pixel is a node and weighted edges are created between each pixel and its 8 neighbors. The weight associated with an edge between two pixels is proportional to the difference of gradient magnitude between these pixels. Optimal paths are computed with the Dijsktra algorithm [Dijkstra, 1959].

An optimal path (which corresponds to a contour) is selected in three steps. The user first has to select a seed point in the image. Then, optimal paths from the user selected seed to all other points in the image are computed. Optimal paths are proposed to the user by moving the mouse cursor. Finally, the user has to select the pixel in the image

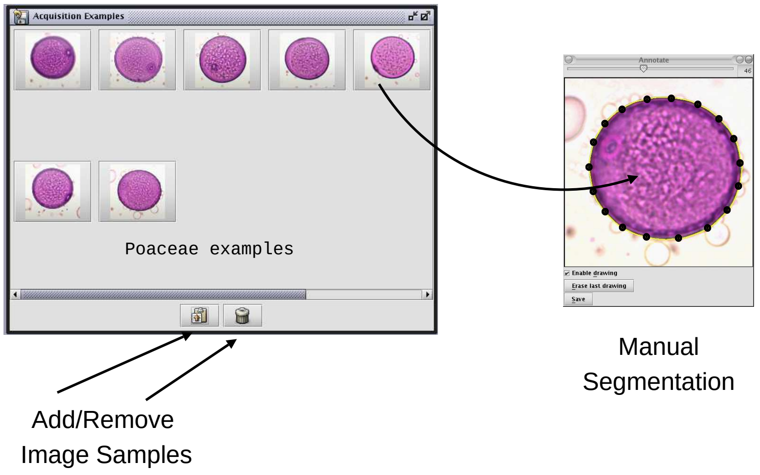


Figure 5.3: Image Samples can be added, removed and selected for manual segmentation.

which is the end of his/her selected boundary. This technique does not require to select manually each pixel of the contour. As seen in fig. 5.4 and in fig. 5.5, the result of manual segmentation is one or several regions which can be annotated by one or several visual concepts used during knowledge acquisition.

In fig. 5.5, it can seen that four regions have been selected by the expert. Region 1 (i.e. a cloudy sky) has been annotated by the visual concepts *Grayish* and *Blue*. The position of this region is *Top*. Region 2 (i.e. the tail) has been described as a *Trapezoid*. Region 3 (i.e. the fuselage) has been annotated by the visual concept *Strong Elongation*. Region 4 (i.e. grass) has been described as *Green*.

In this case, by considering only this image, $Card(\mathcal{AR}) = 6$. The elements of $\mathcal{AR}$ can be found in table 5.1.

i	R_i	C_i
1	r_1	*Grayish*
2	r_1	*Blue*
3	r_1	*Top*
4	r_2	*Trapezoid*
5	r_3	*Strong Elongation*
6	r_4	*Green*

Table 5.1: Annotations of the regions associated with the left image of fig. 5.5

Intelligent scissors are well-adapted for achieving object isolation in a still image. For easy segmentation of objects appearing in image sequences (e.g. video sequences, 2-D slices of 3-D objects), the propagation of the initial contour is required. We use *B-splines snakes* [Brigger et al., 1998] for that purpose. B-snakes are characterized by the following

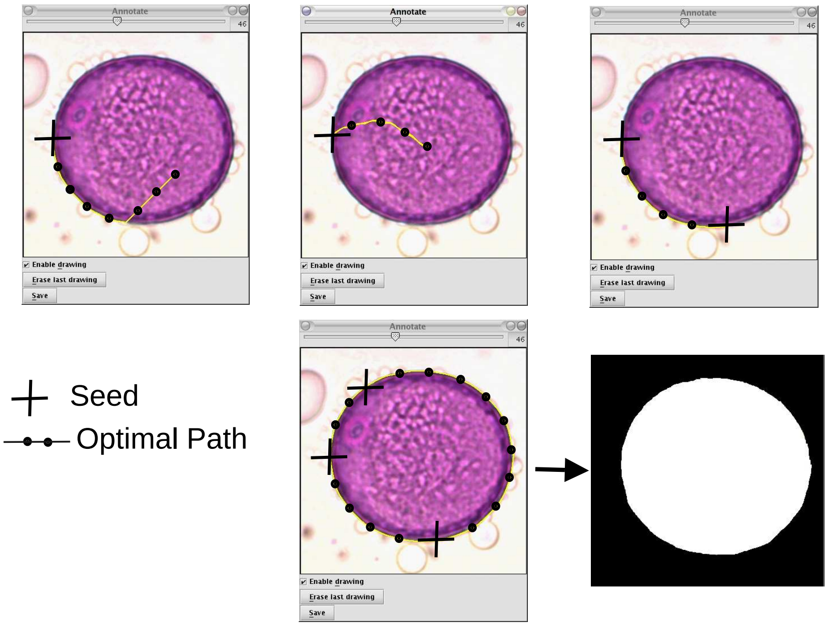


Figure 5.4: Application of the intelligent scissors. The first row of this figure shows three optimal paths computed from the initial seed and proposed to the user. Each proposition is displayed in real-time. The last image of the first row shows the optimal path selected by the user. On the second row, it is shown how an accurate region segmentation has been obtained by selection of three seeds.

points: they can be computed efficiently, few parameters have to be set up, and the smoothness is implicitly taken into account into the model. In addition, B-snakes naturally permit the local control of the curve by controlling individual control points. An overview of existing snake models can be found in [Dumitras and Venetsanopoulos, 2001]. Many models require much computation time and are not suited to user-friendly interaction.

A two-dimensional B-spline curve is defined by its control point as:

$$\mathbf{s}(t) = (s_x(t), s_y(t)) = \sum_{k \in Z} \mathbf{c}(k).\beta^n(t-k)\ ,\ 0 \leq t \leq t_{max} = N-1 \tag{5.1}$$

where $s_x(t)$ and $s_y(t)$ are the x and y spline components, respectively, both parametrized by the curvilinear variable t. N denotes the number of control points, which corresponds to the number of primary B-spline coefficients, denoted by $\mathbf{c}(k) = (c_x(k), c_y(k))$. The energy term associated with a control point is defined as:

$$E(\mathbf{c}(k)) = \sum_{i=0}^{N-1} g(s_x(i), s_y(i)) \tag{5.2}$$

where, $g(x, y)$ is the gradient magnitude at the coordinates (x, y) of the image. The gradient magnitude is computed by using the recursive implementation of the filter pro-

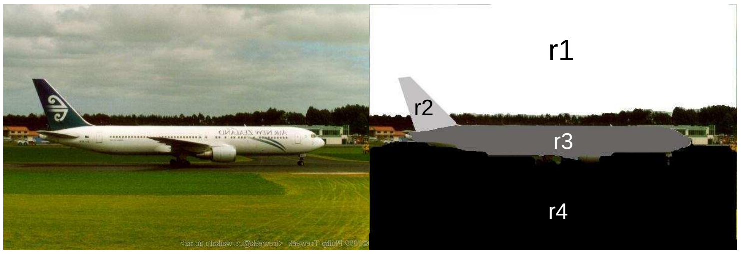


Figure 5.5: This figure shows a training image on the left. On the right, its associated manual segmentation is shown. In this case, four regions of interest have been chosen by the expert.

posed in [Deriche, 1990]. This filter is formaly defined in the one-dimensional case in eq. (5.3) and (5.4). This filter is separable and thus can be applied horizontally and then vertically.

$$f(x) = -kxe^{-\alpha|x|} \tag{5.3}$$

$$k = \frac{(1-e^{\alpha})^2}{e^{-\alpha}} \tag{5.4}$$

α defines defines the width of the filter. A trade-off between the quality of detection and the quality of localization of the edges has to be found by adjusting α.

For a sequence of l images $\{I_0, ..., I_l\}$, the goal of contour propagation is to obtain a set of contours $\{s_0(t), ..., s_l(t)\}$. The computation of $s_i(t)$ from $s_{i-1}(t)$ is achieved by sequential maximization of the energy term E along the contour normal direction at each control point. The location of the maximum energy along the normal direction gives the new location of the control point. The length of the normal used for finding the maximum is a priori defined. This process is depicted in fig. 5.6.

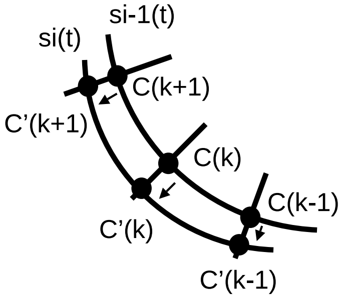


Figure 5.6: The contour $s_i(t)$ is obtained from $s_{i-1}(t)$ by maximizing the contour energy sequentially for each control point. These maximization processes use the normal directions at each control points and lead to $c'(k-1)$, then to $c'(k)$ and finally to $c'(k+1)$. The B-spline $s_i(t)$ is computed from these new control points.

Most of the similar approaches start from a rough segmentation around the object of interest in I_0, then a maximization process on this initial image leads to the initial contour $s_0(t)$. In many cases (e.g. cluttered images), the resulting initial contour is not satisfying and makes the contour propagation difficult. In our case, the initial contour is obtained by using the intelligent scissors. This does not require too much effort from the end-user and makes the contour propagation process more efficient than with an initial rough segmentation. An example of such contour propagation process can be found in fig. 5.7.

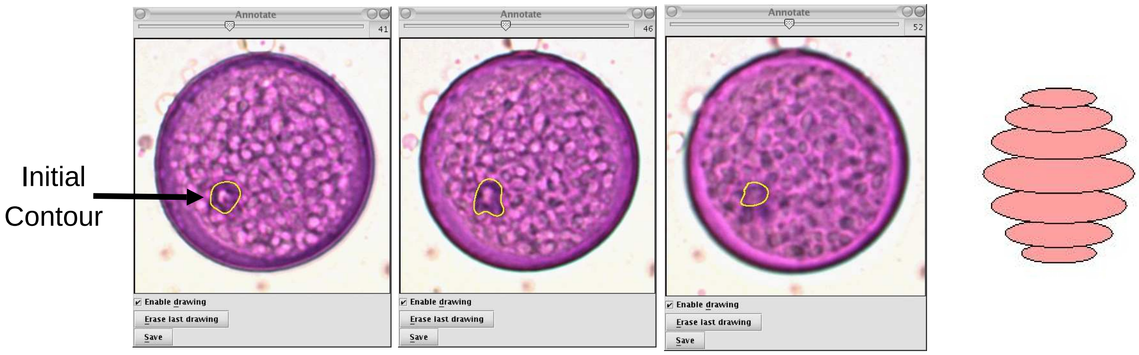


Figure 5.7: This example shows a B-snake evolving through the 2-D slices (represented on the right) of a 3-D pollen grain viewed in a light microscope. The pori (subpart of the grain) has been initially segmented by using the intelligent scissors. The initial contour is then automatically propagated through slices by adapting itself to the pori morphology.

The intelligent scissors bring a real improvement on a completely manual approach where each pixel of a contour has to be selected. It enables easy segmentation of complex objects even in a cluttered background (fig. 5.5). The original combination of this technique with a B-snake approach enables contour propagation in image sequences.

Our approach could be improved by integrating the work presented in [Precioso et al., 2005] where *smoothing* B-Splines are used for reducing the sensitivity to noise. Another interesting point is that topological changes are also taken into account by this approach.

Even if segmentation is made easier than completely manual approach, it remains a tedious task. We have come to the conclusion that complementary approaches had to be used to obtain a large number of annotated samples.

5.2.2 Three-dimensional Model Based Approach

Three-dimensional (3-D) complex objects have many different typical appearances. Obtaining a large quantity of representative samples (e.g. from different points of view) for this kind of objects is a very tedious task. In some domains of interest, it is possible to obtain 3-D models of the objects of interest (e.g. aircrafts, cars, ships). The way these models are used is depicted in fig. 5.8. A 3-D model viewer has been integrated in OntoVis. This viewer is based on the Xj3D project [1] of the Web3D Consortium Working Group. This group is focused on creating a toolkit for the standard format VRML (Virtual Reality Modeling Language) for the Java programming language. The models we use (e.g. aircraft model) are meshes stored in the VRML format.

This viewer enables an expert of the application to associate a model to each domain class. Once, a model as been associated with a domain class, the expert can interact with the 3-D model viewer in order to obtain 2-D projections. User interactions lead to four kinds of transformations: *rotation* around the x, y, z axes, *translation* along the x, y, z axes, *scaling* and finally *projection* in the image plane which is used to obtain the 2-D projections. Resulting 2-D projections can then be used for annotation purposes.

This approach is particularly well adapted for obtaining samples of manufactured objects for which a 3-D model exists. We have used this approach for obtaining annotated regions obtained by the projection of aircraft meshes and then by the annotation of the resulting projections. As seen in fig. 5.8, projections are obtained easily for completely different points of view.

For natural objects like pollen grains, this approach cannot be directly adapted. Natural objects are often deformable and it is difficult to obtain representative models of these objects. Note that interesting work on the image formation model of 3-D translucent object observed in light microscopy can be found in [Dey et al., 2002]. This work should be investigated in order to enable the generation of the 2-D appearance of 3-D translucent objects.

5.2.3 Weakly-Supervised Approach

In sections 5.2.1, it is shown how region annotation by visual concepts is achieved manually. In section 5.2.2, 3-D models are used to obtain 2-D masks which are then annotated by visual concepts. A third approach based on *unsupervised* machine learning techniques is explored in this section. In this case, the region annotation process is divided into the following steps : *automatic segmentation*; *feature extraction*; *clustering and cluster visualization and annotation* (fig. 5.9).

Step 1. All the images of the image training set are segmented into a set of nr regions $\{R_j\}$, with $1 \leq j \leq nr$. A meanshift segmentation algorithm [Comaniciu and Meer, 2002] is used for that purpose. Once the segmentation process is over, the sequence composed of steps (2),(3) and (4) is run for each attribute $a \in \mathcal{A}$ used during knowledge acquisition (i.e. $\exists \mathcal{A}_\alpha$ so that $a \in \mathcal{A}_\alpha$).

Step 2. a be the current considered element of $\mathcal{A}$. A set of feature vectors $\{\mathbf{x_j}\}$ is computed by feature extraction applied to all the regions of $\{R_j\}$. Feature extraction result depends on the features associated with a. For example, if $a = Hue$, a color coherence vector is computed for each R_j. Feature extraction results in a set of couples $\{(R_j, \mathbf{x_j})\}$ where $\mathbf{x_j}$ is the feature vector extracted from R_j.

[1] http://www.xj3d.org

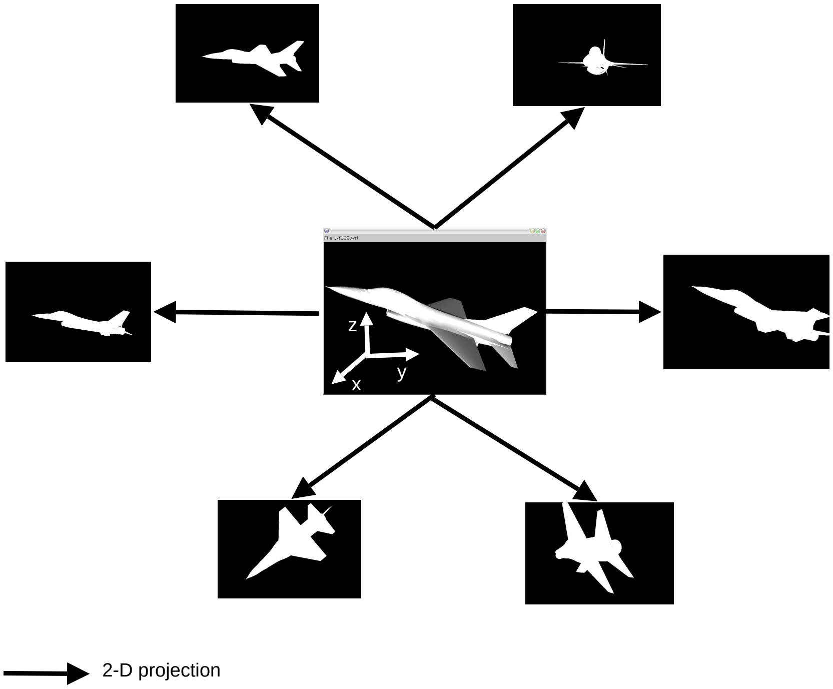


Figure 5.8: From a 3-D model to 2-D projections. This method is convenient to obtain samples of the appearance of a three-dimensional objects. 2d projections are obtained after *rotation*, *translation* and *scaling* of the 3-D model. In this example, the 3-D model is in the middle of the image.

Step 3. The clustering algorithm (e.g. k-means) is applied on $\{\mathbf{x_j}\}$. The result of this clustering is a set of triples $\{(R_j, \mathbf{x_j}, k_j)\}$. $k_i \in \mathbb{N}$ is the numerical label associated with $\mathbf{x_j}$ and R_j. The choice of the number of clusters is currently an issue. Our experiments have shown that 15 clusters is user-friendly starting basis which does not lead to too many clusters. This number can changed by cluster purification.

Step 4. The cluster visualization and annotation step enables the user to assign a symbolic label in terms of visual concepts to the resulting clusters. The k^{th} resulting cluster is visualized by displaying the subset of $\{R_j\}$ labeled by k. The output of cluster visualization and annotation is a set of regions annotated by visual concepts, $\mathcal{AR} = \{(R_i, C_i)\}$. Note that one cluster (and the regions contained in this cluster) can be annotated with several visual concepts. This implies that in general $Card(\mathcal{AR}) \neq nr$.

During this interactive process, impure clusters may be obtained. By an impure cluster we mean that this cluster results from regions representative of several visual concepts. In this case, the clustering algorithm can be reapplied on this cluster in order to improve its purity. For instance, a cluster containing both *Smooth* and *Granulated* regions has to be splitted in two subsets in order to obtain representative samples of these visual concepts. Cluster purity is currently evaluated visually by the end-user. Cluster purification is illustrated in fig 5.10.

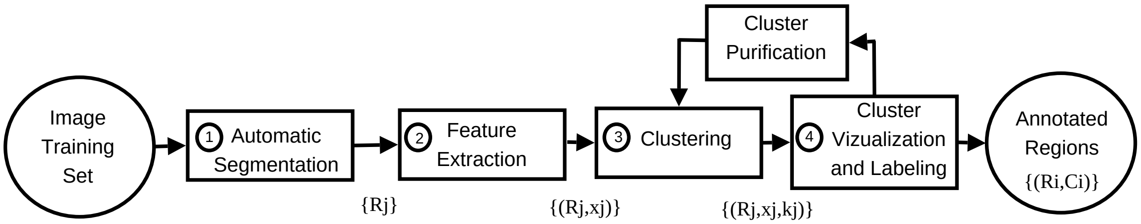


Figure 5.9: From images to annotated image regions. One execution of the sequence composed of steps (2),(3) and (4) corresponds to one visual concept of $\mathcal{A}$. Each visual concept $a \in \mathcal{A}$ is associated with different features $\mathcal{F}(a)$. Depending on the considered visual concept of $\mathcal{A}$, feature extraction and clustering lead to different types of clusters (e.g. clusters resulting from regions of similar hue or of similar size).

An example of use of this weakly-supervised approach is depicted in fig. 5.11. In this case $a = Size$ and $\mathcal{F}(a) = \{area, perimeter\}$. This implies that the extracted features are *area* and *perimeter*. Ten regions are annotated with two visual concepts (i.e. *LargeSize* and *SmallSize*) without performing region segmentation by hand.

This approach is well adapted when the automatic segmentation of the objects of interest contained in the training images is not too difficult. In some application domains, the conditions of image acquisition are well-controlled (e.g. simple background) and reliable segmentation is possible.

5.3 Visual Concept Learning

Section 5.2 has shown how samples of visual concepts used during knowledge acquisition can be obtained by three different complementary approaches: a manual approach, a three-dimensional model based approach and a weakly-supervised approach. In any case, image samples segmentation and annotation leads to a set of annotated region $\mathcal{AR}$.

The goal of visual concept learning is to obtain a set of visual concept detectors for enabling visual detection in an image. This is a problem of generalization of the annotations obtained from image samples segmentation and annotation. We propose to use machine learning techniques to cope with this problem and to produce a set of visual concept detector $\{d_{C_i}\}$. For a feature vector $\mathbf{x} \in \mathbb{R}^n$ computed from a region R and a visual concept C, $d_C(\mathbf{x}) \in [0, 1]$ is confidence degree associated with the hypothesis : "R is a representative sample of C".

This section shows how a visual concept detector dedicated to the detection of a visual concept C is based on the training of a Support Vector Machine (SVM) with positive and negative samples of C.

As seen in fig. 5.12, the input of the visual concept learning process is the set of annotated region $\mathcal{AR}$. Visual concept detectors are obtained by *feature extraction*, *training set building*, *feature selection*, and *training*. We use the hierarchical structure of the visual

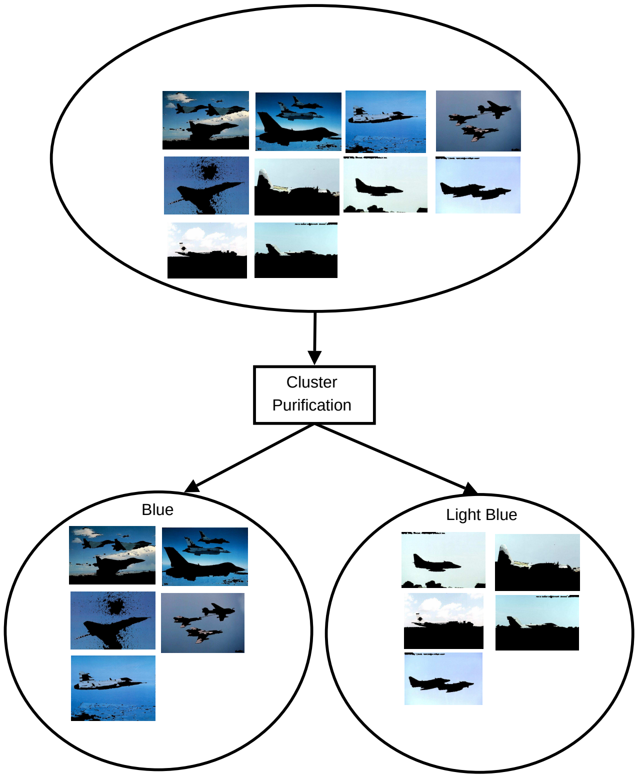


Figure 5.10: Illustration of cluster purification. After automatic segmentation and clustering (for $a = Hue$), a cluster representative of several visual concepts is obtained. The purification of this cluster results in two clusters. The first one is annotated as *Blue*, the second one is annotated as *Light Blue*. This process enables to obtain samples of more precise visual concepts.

concept ontology to obtain simple (i.e. the number of classes in the classification problem is reduced) and focused classification problems.

Feature extraction consists of extracting a feature vector for each annotation performed during image samples segmentation and annotation. For a region R annotated by a visual concept C, features extraction consists of computing the value of the set of numerical features $\mathcal{F}(C)$. The result of feature extraction is a set feature vectors labeled by visual concepts noted X. The goal of *training set building* is, for each visual concept C_i, to produce a set of feature vectors labeled by $+1$ or -1. Each training set is noted T_i.

The role of *feature selection* is to reduce the dimension of each T_i by Linear Discriminant Analysis (LDA) for producing T_i'. Finally comes *training* which produces a visual concept detector by training a SVM using each T_i'.

The main algorithm is given in algorithm 1. Each of these steps is detailed in the next subsections. It sequentially performs feature extraction, training set building, feature selection and training. This algorithm has two input parameters. The first one is the set of image regions labeled by visual concepts ($\mathcal{AR}$). The second one is a visual concept attribute $a \in \mathcal{A}$ (e.g. *Hue*, *Size*, *Geometry*) which defines the category of visual concepts which

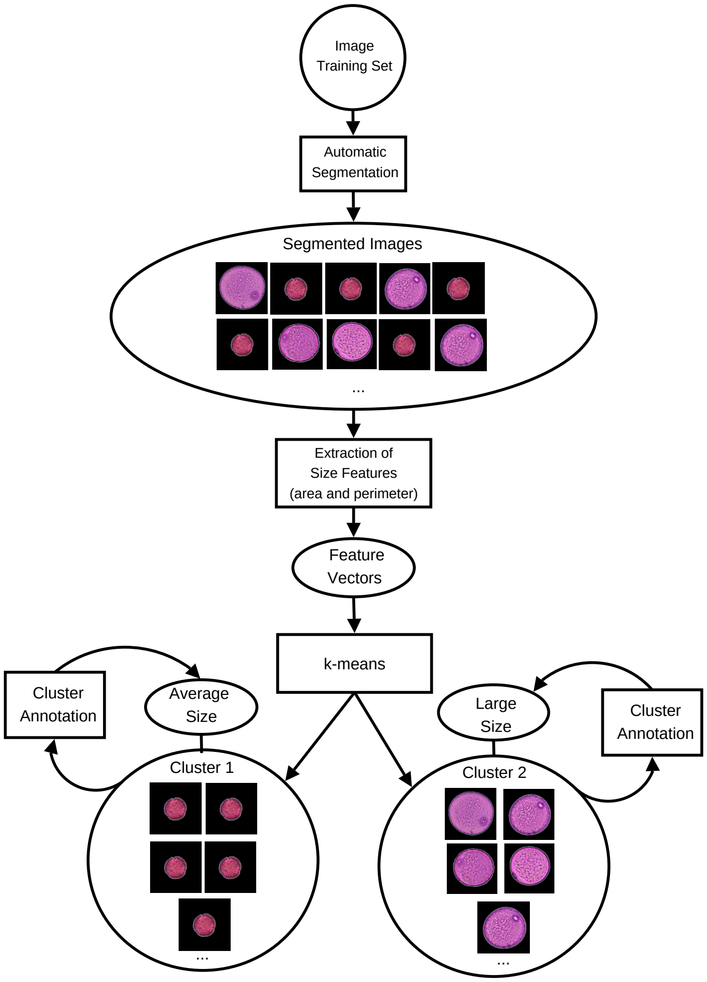


Figure 5.11: An example of weakly-supervised annotation of some pollen grain images. The input of the process is the image training set. After automatic segmentation and clustering by *k-means*, two clusters are obtained. In this case, the expert uses the visual concept $AverageSize$ as an annotation for cluster 1 and $LargeSize$ as an annotation for cluster 2.

has to be learned. For learning all the visual concepts used during region annotation, this algorithm has to be run n times (with $n = Card(\mathcal{A})$) with the second parameter sequentially set to each element of $\mathcal{A}$ (e.g. $Elongation$, $Pattern$, $Brightness$). The result of algorithm 1 is a set of detectors trained to the detection of C and its children.

5.3.1 Feature Extraction

Feature extraction uses the set of the annotated regions $\mathcal{AR}$ to produce a set of numerical vectors $\mathbf{x}$ labeled by visual concepts. Algorithm 2 gives a detailed description of feature extraction. For a visual concept C and an image region R, the $extractFeatures$ function assigns a value to the features associated with C (i.e. $\mathcal{F}(C)$). For instance, $\mathcal{F}(Size) = \{area, perimeter\}$. By considering a region R, $extractFeatures(R, F(Size))$ computes

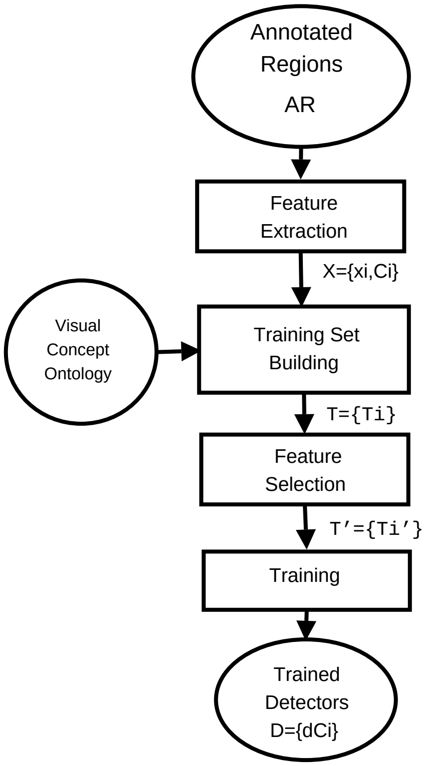


Figure 5.12: Visual concept learning. The annotated regions are obtained by image samples segmentation and annotation. The result of visual concept learning is a set of visual concept detectors.

Algorithm 1 $VisualConceptLearning(\mathcal{AR}, C)$

```
  $X \leftarrow \{\}$
  $T \leftarrow \{\}$
  $T' \leftarrow \{\}$
  $D \leftarrow \{\}$
  $X \leftarrow FeatureExtraction(\mathcal{AR}, C)$
  $T \leftarrow TrainingSetBuilding(X, C)$
  $T' \leftarrow FeatureSelection(C, T)$
  $D \leftarrow Training(C, T')$
  $return\ D$
```

the area and perimeter of R. If the region is annotated by the visual concept $LargeSize$, a feature vector $\mathbf{x} \in \mathbb{R}^2$ is associated with the visual concept $LargeSize$.

5.3.2 Training Set Building

The visual concept detection problem is seen as a two class decision problem. We use binary classifiers (e.g. SVM) for achieving visual concept detection. Training of such binary classifiers is done with numerical feature vectors labeled by $+1$ or -1. X is a set of feature vectors labeled by visual concepts and does not meet this requirement: X is a

Algorithm 2 $FeatureExtraction(\mathcal{AR}, C)$

```
for all $(R_i, C_i) \in \mathcal{AR}$ do
  if $C_i \preceq_\Theta C$ then
    $\mathbf{x} \leftarrow extractFeatures(R_i, \mathcal{F}(C))$
    $X \leftarrow X \cup (\mathbf{x}, C_i)$
  end if
end for
return $X$
```

set of feature vectors labeled by visual concepts.

The goal of training set building is to produce a training set T_i associated with each visual concept C_i from X. A training set T_i is a set of labeled feature vectors of $\mathbb{R}^n$ labeled by $y \in \{-1, 1\}$. $y = 1$ means that $\mathbf{x_i}$ is a representative sample of C_i. $y = -1$ means that $\mathbf{x_i}$ is a negative sample of C_i.

We use the hierarchical structure of the ontology to obtain simpler (the number of classes in the classification problem is reduced) and focused classification problems. For instance, if the geometry of a region is classified as a polygonal surface, the refinement of this classification (e.g. trapezoid surface, quadrilateral surface) does not need to take into account elliptical surfaces.

Positive samples of C_i are samples labeled by C_i. Negative samples of C_i are positive samples of brothers of C_i. The set of samples labeled explicitly by $Not\ C_i$ is also added to the set of negative samples of C_i.

P_i is the set of representative training vectors of a visual concept C_i. N_i is the set of training vectors computed on negative samples of a visual concept C_i.

$$\begin{cases} P_i = \{(\mathbf{x_j}, +1) \mid \exists(\mathbf{x_j}, C_j),\ C_j \preceq_\Theta C_i\} \\ N_i = \{(\mathbf{x_j}, -1) \mid \exists(\mathbf{x_j}, C_j),\ (C_j \preceq_\Theta (C_k \in brothers(C_i))\ \vee\ C_j = Not\ C_i) \wedge (\mathbf{x_j}, +1) \notin P_i\} \\ T_i = P_i \cup N_i \end{cases}$$

5.3.3 Feature Selection

The goal of feature selection is to reduce the dimensionality of each T_i. The dimension of each T_i is n. A well known method dedicated to this purpose is Principal Component Analysis (PCA) which is an unsupervised technique and as such does not include label information of the data. In our case case, for a given visual concept, all feature vectors are labeled by $+1$ or -1: we are in a supervised case. This implies that the Fisher Linear Discriminant Analysis (LDA) is well adapted. This method selects features that maximize the ratio of the *between-class scatter* to the *within-class scatter*.

Let the between-class scatter be defined as

$$S_b = \sum^{\Omega_i} P_{\Omega_i}(\mu_{\Omega_i} - \mu)(\mu_{\Omega_i} - \mu)^T \tag{5.5}$$

where $\{\Omega_i\}$ is a set of classes. In our case, the number of classes is 2 (i.e. positive and negative samples of each visual concept). μ is the mean of all samples and μ_{Ω_i} is the mean of class Ω_i with prior probability P_{Ω_i}. And the within-class scatter matrix is defined as:

$$S_w = \sum^{\Omega_i} P_{\Omega_i} S_{\Omega_i} \tag{5.6}$$

with

$$S_{\Omega_i} = E[(\mathbf{x} - \mu_{\Omega_i})(\mathbf{x} - \mu_{\Omega_i})^T \mid \mathbf{x} \in \Omega_i] \tag{5.7}$$

The dimension of the input space is $n : \mathbf{x} \in \mathbb{R}^n$. If S_w is non-singular, the optimal projection W_{opt} is chosen as a matrix with orthonormal columns which maximize the ratio of the determinant of the between-class scatter matrix to the determinant of the within-class scatter matrix:

$$J(W_{opt}) = \arg\max_W \frac{|W^T S_b W|}{|W^T S_w W|} \tag{5.8}$$

W_{opt} is a $m \times n$ $(m \leq n)$ matrix.

A solution to the optimization problem of eq. (5.8) is to solve the generalized eigen value problem:

$$S_b W_{opt} = \lambda S_w W_{opt} \tag{5.9}$$

The most frequently used LDA algorithm in practice is based on simultaneous diagonalization. The basic idea is to find W_{opt} such that W_{opt} simultaneously diagonalizes both S_w and S_b:

$$W_{opt} S_w W_{opt}^T = I \textit{ and } W_{opt} S_b W_{opt}^T = \Lambda \tag{5.10}$$

For a training set T_i (i.e. positive and negative samples of C_i), the optimization problem defined by eq. (5.8) is solved and the first $m \in \mathbb{N}$ rows of W_{opt} which correspond to the m largest eigen values of Λ are used as a transformation matrix M_{C_i} $(m \times n)$. $m \leq n$ is the new dimension of the training set. As seen in algorithm 1 the set of reduced training sets T' is produced by the by the $FeatureSelection$ function with C and T for parameters. The resulting set of reduced training sets T' is defined as:

$$\begin{cases} T' = \{T'_i\} \\ T'_i = (M_{C_i} T_i^T)^T \\ \forall C_i, C_i \preceq_\Theta C \end{cases}$$

5.3.4 Training

A visual concept detector d_{C_i} is associated with each concept C_i. As seen in table 5.2, several types of decision can be inferred from $d_{C_i}(\mathbf{x})$ The distance reject notion has been introduced in [Dubuisson and Masson, 1993] in a formal probabilistic framework. The distance reject notion enables to tackle classification problems with incomplete knowledge about classes. In probabilistic terms, the distance reject case takes into account the regions of a parametric space where the density of probability is lower than a threshold. In our case, we have used the notion of distance reject in order to be able to decide if some visual concepts are missing in the visual concept ontology. If at a given level of abstraction in the ontology, the decision reject decision is often taken, it may mean that an unknown type of visual concept is often appearing and thus that the ontology has to be completed. The ambiguity reject case takes is useful when the confidence related to several decisions is greater than a threshold. These notions are illustrated in fig. 5.13.

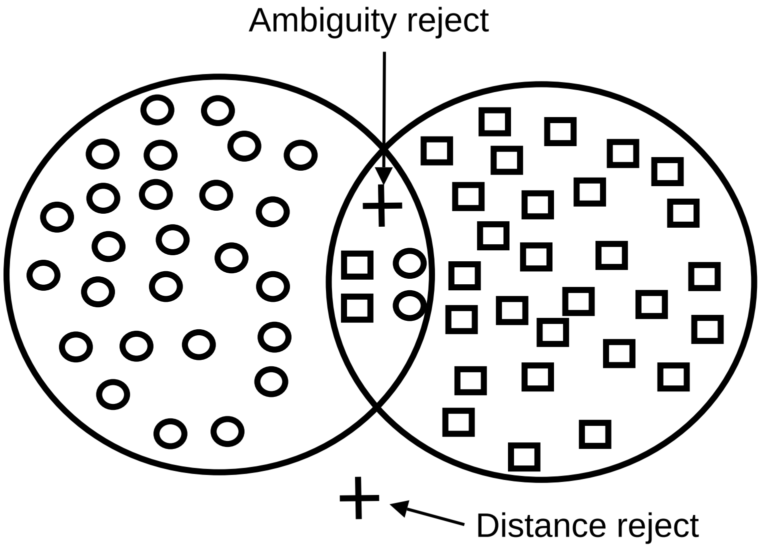


Figure 5.13: Illustration of the cases of *ambiguity reject* and *distance reject* for a two class decision problem. Classifying an element consists of assigning it to the class *square* or to the class *circles*. The *ambiguity reject* case corresponds to a region of the parametric space where the two classes are overlapping. The *distance reject case* corresponds to a region of the parametric space which is too distant from the two classes.

We define two thresholds $amb_{th} \in]0.5, 1[$ and $dist_{th} \in]0, 1[$. amb_{th} is the ambiguity reject threshold and defines the degree of confidence needed to take the decision of detecting a concept. $dist_{th}$ is the distance reject threshold.

We use Support Vector Machines (SVM) for obtaining the set of visual concept detectors. SVM training consists of finding a hyper-surface in the space of possible inputs (i.e. feature vectors labeled by $+1$ or -1). This hyper-surface will attempt to split the positive examples from the negative examples. The split will be chosen to have the largest distance

Decision	Definition
C_i detected	$d_{C_i}(\mathbf{x}) \geq amb_{th}$
C_i not detected	$d_{C_i}(\mathbf{x}) \leq 1 - amb_{th}$
Ambiguity reject	$d_{C_i}(\mathbf{x}) \in]1 - amb_{th}, amb_{th}[$
Distance reject	$\forall C_j \ \preceq_\theta \ a \mid a \in \mathcal{A} \ and \ C_i \preceq_\theta a, \ d_{C_j}(\mathbf{x}) \leq dist_{th}$

Table 5.2: Decision types inferred from $d_{C_i}(\mathbf{x}) \in [0,1]$. $\mathbf{x} \in \mathbb{R}^m$ is a feature vector extracted from an image region.

from the hyper-surface to the nearest of the positive and negative examples. Intuitively, this makes the classification correct for testing data that are near, but not identical to the training data (fig. 5.14). More information can be found in [Burges, 1998] and in [Vapnik, 1995].

For learning all the visual concepts, algorithm 1 has to be called with each visual concept $a \in \mathcal{A}$. In this case, for each possible value of a, a visual concept detector will be computed.

Let $(x_i, y_i)_{1 \leq i \leq ne}$ be a set of training examples, each example $\mathbf{x_i} \in \mathbb{R}^m$, m being the dimension of the input space, belongs to a class labeled by $y_i \in \{-1, 1\}$.

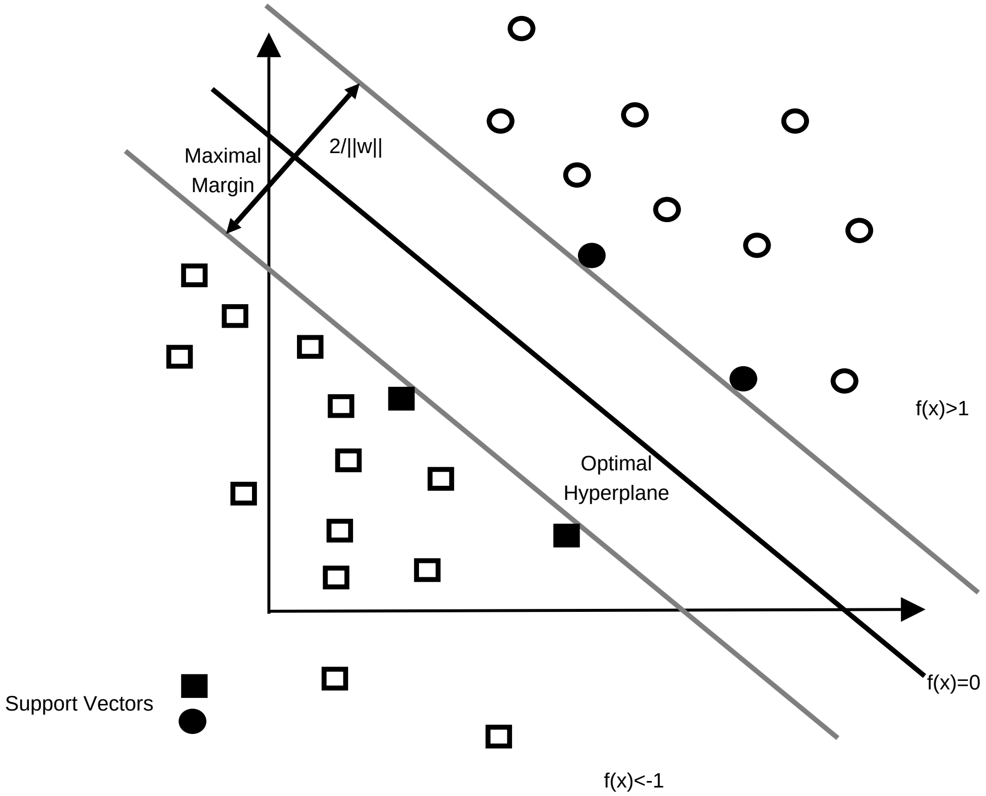


Figure 5.14: The goal of the optimatization process involved in the training of a SVM is to *maximize the margin*. In this case, the training set is *separable*.

The goal is to obtain an hyperplane which divides the set of examples such that all the points with the same label are on the same side of the hyperplane. This can can be achieved by finding $\mathbf{w}$ and b so that,

$$y_i(\mathbf{w} \cdot \mathbf{x_i} + b) > 0, \ i = 1, ..., ne \tag{5.11}$$

If there exists an hyperplane satisfying eq. (5.11), the training set of examples is said to be separable. In this case, it always possible to rescale $\mathbf{w}$ and b so that

$$\arg \min_{1 \leq i \leq ne} y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1, \quad i = 1, ..., ne \tag{5.12}$$

The closest point to the hyperplane has a distance $1/|\mathbf{w}|$. Equation (5.11) becomes:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1 \tag{5.13}$$

Among the separating hyperplanes, the one for which the distance to the closest point is maximal is called the optimal separating hyperplane. Since the distance to the closest point is $1/|\mathbf{w}|$, finding the optimal separating hyperplane is done by minimizing $|\mathbf{w}|^2$ under the constraint (5.13).

The quantity $2/|\mathbf{w}|$ is called the *margin*, and thus the optimal separating hyperplane is the separating hyperplane which maximized the margin. The larger the margin, the better the generalization is expected to be. Since $|\mathbf{w}|^2$ is convex, minimizing the margin under linear constraints (5.13) can be achieved with Lagrange multipliers, it can be shown that this optimization problem is equivalent to the maximization of:

$$W(\alpha) = \sum_{i=1}^{ne} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{ne} \alpha_i \alpha_j y_i y_j \mathbf{x_i} \mathbf{x_j} \tag{5.14}$$

with $\alpha_i \geq 0$ an under constraint $\sum_{i=1}^{n} y_i \alpha_i = 0$. This can be achieved by the use of standard quadratic methods. Once the solution vector $\alpha^0 = (\alpha_1^0, \alpha_2^0, ..., \alpha_{ne}^0)$ of the maximization problem (5.14) has been found, the optimal separating hyperplane $(\mathbf{w_0}, b_0)$ has the following expansion:

$$\mathbf{w}_0 = \sum_{i=1}^{ne} \alpha_i^0 y_i \mathbf{x}_i \tag{5.15}$$

The *support vectors* are the points for which α_i^0 satisfy eq. (5.13). Considering the expansion (5.15) of $\mathbf{w}_0$, the output function f is defined as:

$$f(\mathbf{x}) = \sum_{i=1}^{ne} \alpha_i^0 y_i \mathbf{x}_i \cdot \mathbf{x} + b^0 \tag{5.16}$$

sign$(f(\mathbf{x}))$ is usually used as a binary decision function. If *sign*$(f(\mathbf{x}))$ is positive, resp.

negative this means that the test sample $\mathbf{x}$ belongs to the class of the training samples labeled by $+1$, resp. -1.

This approach can be made non linear via the use of a *kernel* which enables the mapping of the input data into a high-dimensional feature space through some nonlinear mapping chosen a priori [Vapnik, 1995]. The optimal separating hyperplane is constructed in this hyperplane. If $\mathbf{x}$ is replaced by its mapping in the feature space $\Phi(\mathbf{x})$, eq. (5.14) becomes:

$$W(\alpha) = \sum_{i=1}^{ne} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{ne} \alpha_i \alpha_j y_i y_j \Phi(\mathbf{x_i}) \cdot \Phi(\mathbf{x_j}) \tag{5.17}$$

If we have $K(\mathbf{x_i}, \mathbf{x_j}) = \Phi(\mathbf{x_i}) \cdot \Phi(\mathbf{x_j})$, then only K is needed in the training algorithm and the mapping Φ is never explicitely used. Given a symmetric positive kernel $K(\mathbf{x}, \mathbf{y})$, Mercer's theorem indicates [Scholkopf and Smola, 2001] that there exists a mapping Φ such that $K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y})$. Once a kernel K satisfying the condition of Mercer has been chosen, the training algorithm consists of minimizing (5.18).

$$W(\alpha) = \sum_{i=1}^{ne} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{ne} \alpha_i \alpha_j y_i y_j K(\mathbf{x_i}, \mathbf{x_j}) \tag{5.18}$$

and the output function then becomes:

$$f(\mathbf{x}) = \sum_{i=1}^{ne} \alpha_i^0 y_i K(\mathbf{x_i}, \mathbf{x}) + b^0 \tag{5.19}$$

There are different types of kernels. Most commonly used are detailed in table 5.3.

Kernel Type	Definition	Comment
Polynomial	$K(\mathbf{x}, \mathbf{x'}) = (\mathbf{x}.\mathbf{x'} + 1)^p$	p is a priori defined
Radial Basis	$K(\mathbf{x}, \mathbf{x'}) = e^{-\frac{\lvert \mathbf{x}-\mathbf{x'} \rvert^2}{2\sigma^2}}$	σ is a priori defined
Sigmoid	$K(\mathbf{x}, \mathbf{x'}) = tanh(a(\mathbf{x}.\mathbf{x'} - b))$	a and b are a priori defined

Table 5.3: Training a SVM implies choosing a kernel and its parameters.

As explained in [Platt, 2000], the output function defined in eq. (5.19) is uncalibrated and can be *normalized*.

For achieving training, one SVM is trained for each T'_i to obtain each $d_{C_i} : \mathbb{R}^m \rightarrow [0, 1]$. Our experiments has led to the choice of a radial basis kernel with $\sigma = 0.5$. As seen in algorithm 1 the set of visual concept detectors D is produced by the by the $Training$ function with C and T for parameters. The resulting set of visual concept detectors D is defined as:

$$\begin{cases} D = \{d_{C_i}\} \\ \forall C_i, C_i \preceq_\Theta C \end{cases}$$

5.4 Object Learning

In section 5.3, we have presented how visual concept learning is achieved. Performing visual concept learning can also be achieved by using domain classes. This way of operating is more user-friendly because it is done in the terminology of the application domain. Moreover, for a given knowledge base composed of a set of domain classes described by visual concepts, it can be convenient to only take into account the visual concepts describing a subset of the classes. This is an important functionality for developing and grounding knowledge bases in an *incremental* way. For achieving learning, algorithm 1 can be applied for each $a \in \mathcal{A}_\alpha$ and by considering all the annotations ($\mathcal{AR}$). This approach is not always convenient when setting up an object categorization system. Learning algorithms are time consuming and it is very useful to be able to focus on a part of the knowledge base. The proposed object learning algorithm is also convenient for enabling object learning even if the image annotation process is not achieved. Therefore, some objects described by visual concepts for which few or no samples exist can be ignored.

As sketched in fig. 5.15, the learning process of domain object classes is composed of three main steps. The object learning algorithm is initiated by a request containing a domain class $root \in \Phi$, the root class from which visual concepts will be considered as candidates for visual concept learning. The set of attributes to be considered $\mathcal{A}' \subset \mathcal{A}$ can also be defined in the request. The object learning algorithm performs a hierarchical exploration of the knowledge base and collects the visual concepts used as attribute values for describing the domain classes. This step is formally described by algorithm 3. This algorithm returns the set of visual concepts (VC) describing the domain class *root* and its subclasses. The annotations set is then filtered by considering the annotations involving the visual concepts of VC. The result of filtering is a set of annotations $\mathcal{AR}'$ defined as:

$$\mathcal{AR}' = \{(R_j, C_j) \mid (R_j, C_j) \in \mathcal{AR} \wedge C_j \in VC\} \tag{5.20}$$

Finally, the visual concept learning algorithm (algorithm 1) is called for each $a \in \mathcal{A}'$ and by using the filtered annotations (i.e. $VisualConceptLearning(\mathcal{AR}', a)$).

Algorithm 3 $VisualConceptAccumulation(root \in \Phi, \mathcal{A}' \subset \mathcal{A})$

```
for all $\alpha \preceq_\Phi root$ do
  for all $a \in \mathcal{A}' \cap \mathcal{A}_\alpha$ do
    for all $v \in \mathcal{V}(a)$ do
      $VC \leftarrow VC \cup v$
    end for
  end for
  for all $sp \in \mathcal{S}_\alpha$ do
    $VC \leftarrow VC \cup VisualConceptAccumulation(sp, \mathcal{A}')$
  end for
end for
return $VC$
```

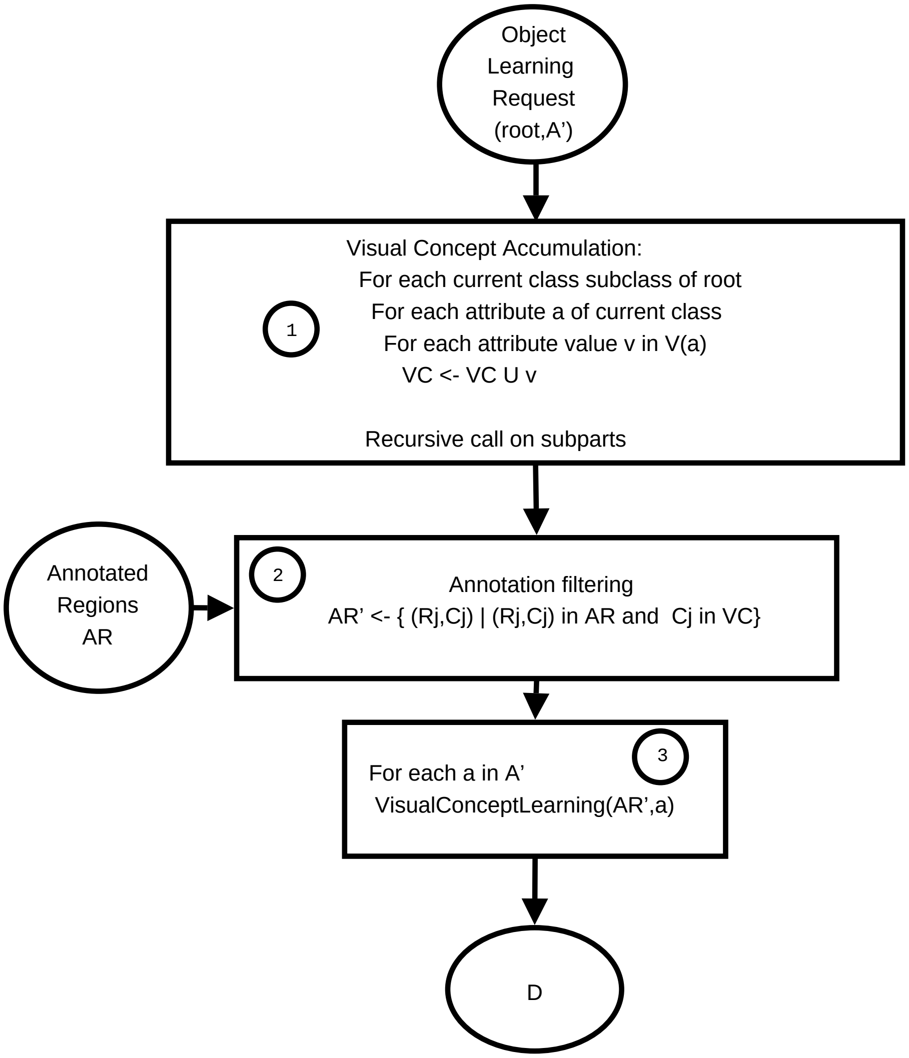

Object Learning Request (root,A')
Visual Concept Accumulation:
For each current class subclass of root
For each attribute a of current class
For each attribute value v in V(a)
VC <- VC U v
Recursive call on subparts
1
Annotated Regions AR
2
Annotation filtering
AR' <- { (Rj,Cj) | (Rj,Cj) in AR and Cj in VC}
3
For each a in A'
VisualConceptLearning(AR',a)
D

Figure 5.15: The object learning algorithm. This algorithm is initiated by an object learning request. The output is a set of visual concept detectors.

5.5 Conclusion

We have presented a learning approach designed for learning domain object classes described by visual concepts. This learning phase is composed of two main steps: *image samples segmentation and annotation* and *visual concept learning*. The goal of visual concept learning is to produce a set of visual concept detectors for each visual concept used during knowledge acquisition. One fundamental aspect of object learning is that it is not designed to learn directly domain classes from image samples. An intermediate level of abstraction, the visual concepts are used. The roles of visual concepts are the following: first, they decompose the problem of object learning into simpler (i.e. lower dimensionality) and meaningful learning problems; second, the low-level image features attached to each visual concept enable meaningful feature extraction (e.g. color features are not extracted for characterizing the geometry of an object).

Image sample segmentation and annotation consists of obtaining a set of image regions annotated by visual concepts. Three different methods for obtaining segmented and annotated image regions are presented. The first method is based on a tool called intelligent scissors which enables easy isolation of an object in any image. We have combined this technique with B-snakes in order to enable contour propagation in spatio-temporal image sequences. The second method uses 3-D models of the objects of interest for producing 2-D projections which can be annotated by visual concepts. The third method uses clustering technique and automatic segmentation techniques. This weakly-supervised approach takes as input a set of training images, automatically segments them and clusters the regions by similarity. The expert has only to label resulting clusters (i.e. several regions at the same time) and not image regions one by one.

Visual concept learning consists of producing a set of visual concept detectors from the annotated regions. For this purpose, the following tasks are achieved: feature extraction, training set building, feature selection and training of the visual concept detectors. For a feature vector extracted from a region of interest and a visual concept C, four decisions can be inferred from the output of a visual concept detector: C detected, C not detected, ambiguity reject and distance reject. Visual concept learning uses the hierarchical structure of the visual concept ontology to obtain well-focused visual concept detectors. This implies that fewer samples have to be taken into account for training each visual concept detector. This also allows the dimensionality of each visual concept detection problem to be reduced.

An object learning algorithm has also been presented. This algorithm enables to perform visual concept learning by taking into account a subset of the domain classes. This functionality is particularly useful for developing an object categorization system in an incremental way.

The next chapter shows how the resulting visual concept detectors enable object recognition.

Chapter 6

Object Categorization

6.1 Introduction

Chapter 4 has shown how knowledge from a domain of interest is acquired as set of domain object classes described by visual concepts. The visual concept layer stands as a user-friendly interface which links domain knowledge to low-level features and algorithms. We have shown in chapter 5 that object learning is achieved through the learning of the visual concepts describing the domain objects. Visual concept learning consists of using segmented and annotated image samples for computing a set of *visual concept detectors*. A visual concept detector can be used to detect a visual concept in any image.

The goal of the current chapter is to show how the acquired knowledge combined with the visual concept detectors are used for enabling object categorization. Object categorization means selecting one or several regions of an image so that they match the visual description, the subpart relations and the spatial relations of at least one class of the domain taxonomy. This is a model-based approach. Object categorization is initiated by a *categorization request* which contains an image. The result of object categorization is one or several regions of the input image associated with compatible domain classes and a visual description in terms of visual concepts. If no region of the input image is compatible with one domain class, then the categorization result is empty.

The key mechanism of object categorization is *hypothesis generation/verification*. As seen in fig. 6.1, the domain knowledge base is involved in the object categorization process. The visual description of each domain class is used as a hypothesis (i.e. an expected set of visual concepts) which has to be verified in the image by *segmentation*, *feature extraction* and by *visual concept detection*. Hypothesis verification consists of evaluating the *compatibility* between the expected visual concepts and the visual concepts detected in the image. If there is compatibility, the current hypothesis is validated and a refined hypothesis is generated by following the specialization links of the domain class hierarchy. The object categorization process is applied recursively for object subparts.

The notion of *unknown object* is used to store the visual concepts detected by visual concept detection so as to ease compatibility evaluation. The unknown object stands as a *fact base* used for evidence accumulation.

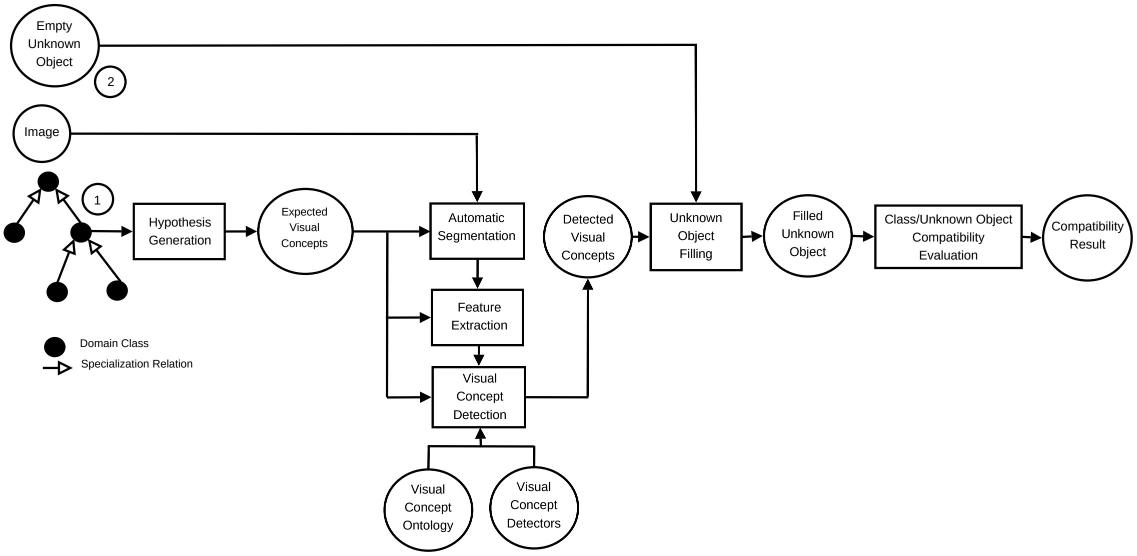


Figure 6.1: Illustration of how the compatibility between a domain class (1) and an unknown object (2) is achieved. The visual description of each class of the domain class hierarchy is used for generating a hypothetical visual description. The image to be interpreted is then *segmented*. Low-level features are *extracted* from the resulting regions. The set of *visual concept detectors* obtained during object learning uses the extracted features to produce a set of *detected visual concepts*. The unknown object has to be filled with the detected visual concepts before evaluating its compatibility with the current class.

Section 6.2 provides a detailed description of the main categorization algorithm. The structure of a categorization request that initiate the categorization is first detailed. This section also shows how a hypothetical visual description is generated from a domain class as a set of expected visual concepts. Then, each step of object categorization is presented: image segmentation, feature extraction and visual concept detection. We also detail how compatibility between the unknown object and the classes of the domain is computed. A presentation of the output of the object categorization algorithm is also proposed: it is shown that object categorization is explicit and is performed in the terminology of the domain of expertise and also in terms of visual concepts. **Section 6.3** provides a conclusion and a discussion of our approach of object categorization.

6.2 Categorization Algorithm

The categorization algorithm is first initiated by a *categorization request*. As seen in fig. 6.2, the categorization algorithm performs a hierarchical exploration of the domain taxonomy. Based on the visual description of the explored domain classes, hypotheses are generated as sets of *expected visual concepts*. An *unknown object* is used to hold the visual concepts detected in the image. This visual object stands as a fact base. The unknown object is filled with the detected visual concepts by segmentation, feature extraction and visual concept detection. At each step of the hierarchical exploration of the domain taxonomy, the compatibility between the visual object and the current class is computed. If compatibility holds, then the hypothesis is refined.

A simplified version of this algorithm can be found in fig. 6.2. A detailed description of this algorithm can be found in algorithm 4. The object categorization algorithm has four input parameters: the root class α_r, the image I, the compatibility threshold $comp_{th}$, and the segmentation fact base which is used to store regions associated with unknown objects. The segmentation fact base has to be empty before the initialization of object categorization.

This section aims at explaining how these input parameters are used by the underlying mechanisms of this algorithm.

6.2.1 Categorization Request

A categorization request initiates categorization. It contains the image to interpret, a compatibility threshold $comp_{th}$ which defines the degree of confidence required for considering a domain class as recognized. Another element of the categorization request is the domain class α_r used as a starting point for categorization. Setting α_r is useful for considering a subset of the domain classes defined as $\{\alpha \in \Phi \mid \alpha \preceq_\Phi \alpha_r\}$. Any domain class not in this set is not considered during the object categorization process. This implies that the image to interpret cannot be interpreted in terms of the ignored classes. Categorization requests depend on the context of use of the system and are more or less abstract depending of the level of expertise of the user. The hierarchical structure of domain knowledge combined with the possibility of setting the root class in the categorization request is particularly useful for addressing categorization requests at different levels of abstraction.

For instance, by using a system specialized in the categorization of vehicles, an expert may be interested only in the objects belonging to the class of *motorbikes* and not belonging to the class of *cars*. This information can be integrated in the categorization request so that the categorization process only takes into account the class of motorbikes and by ignoring the class of cars. Figure 6.3 shows a categorization process started at an intermediate level of abstraction of a hierarchy of domain classes.

6.2.2 Unknown Object Initialization

The first step of the object categorization algorithm is to initialize the unknown object. An unknown object is used to store visual concepts detected in the image and is also used for computing the compatibility with the domain classes. The unknown object associated with a domain class reflects its structure. More precisely, all the attributes (i.e subpart attributes, visual attributes, spatial relation attributes) are common to the class and to its associated unknown object. The value of these attributes is used to store the evidence detected in the image. An unknown object $\mathcal{O}$ is composed of the following elements:

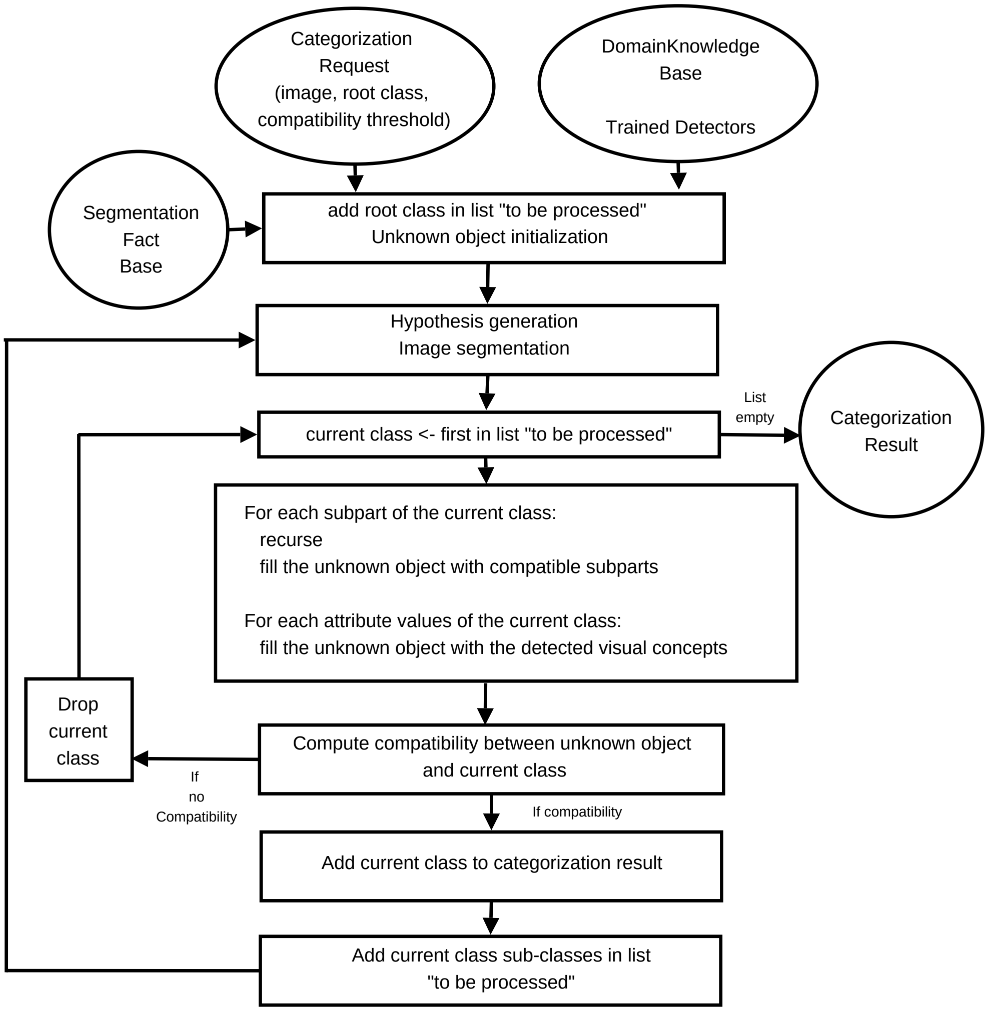

Categorization
Request
(image, root class,
compatibility threshold)
DomainKnowledge
Base
Trained Detectors
Segmentation
Fact
Base
add root class in list "to be processed"
Unknown object initialization
Hypothesis generation
Image segmentation
List
empty
Categorization
Result
current class <- first in list "to be processed"
For each subpart of the current class:
recurse
fill the unknown object with compatible subparts
For each attribute values of the current class:
fill the unknown object with the detected visual concepts
Drop
current
class
Compute compatibility between unknown object
and current class
If
no
Compatibility
If compatibility
Add current class to categorization result
Add current class sub-classes in list
"to be processed"

Figure 6.2: Simplified version of the object categorization algorithm. This algorithm is initiated by a categorization request. It uses the acquired domain knowledge and the trained detectors. The different steps of the object categorization algorithm are: unknown object initialization, hypothesis generation, image segmentation, unknown object filling with the detected visual concepts, recursion of the categorization process for the subparts, and hypothesis refinement.

Algorithm 4 $ObjectCategorization(\alpha, I, comp_{th}, segmentationFactbase)$

```
$list \leftarrow \{\alpha\}$
$\mathcal{O} \leftarrow initializeUnknownObject(\alpha)$
while $isEmpty(list) = false$ do
  $\beta \leftarrow getFirst(list)$
  $\mathcal{H} \leftarrow generateHypothesis(\beta)$
  $R \leftarrow segmentation(\mathcal{H}, I)$
  {Recurce on subparts}
  for all $sa \in \mathcal{S}_\beta$ do
    $spRes \leftarrow \{\}$
    for all $sp \in \mathcal{V}_\beta(sa)$ do
      $spRes \leftarrow spRes \cup ObjectCategorization(sp, R, comp_{th}, segmentationFactBase)$
      $\mathcal{V}_\mathcal{O}(sa) \leftarrow \mathcal{V}_\mathcal{O}(sa) \cup spRes$
    end for
  end for
  {Fill unknown object with detected visual concepts and subparts}
  $\mathcal{O} \leftarrow unknownObjectFilling(R, \mathcal{H}, \mathcal{O})$
  {Evaluate compatibility between visual object and current class}
  $\mathcal{O}_{compatibility} \leftarrow \frac{1}{3} \times (visualCompatibility(\mathcal{O}, \beta) + subpartCompatibility(\mathcal{O}, \beta) + spatialRelationCompatibility(\mathcal{O}, \beta, segmentationFactBase))$
  {Explore subclasses}
  if $\mathcal{O}_{compatibility} \geq comp_{th}$ then
    $res \leftarrow res \cup \{\mathcal{O}\}$
    $segmentationFactBase \leftarrow segmentationFactBase \cup (\mathcal{O}, R)$
    if $hasChildren(\beta) = true$ then
      for all $\gamma \in Children(\beta)$ do
        $list \leftarrow \{\gamma\} \cup list$
      end for
    end if
  end if
end while
$return\ res$
```

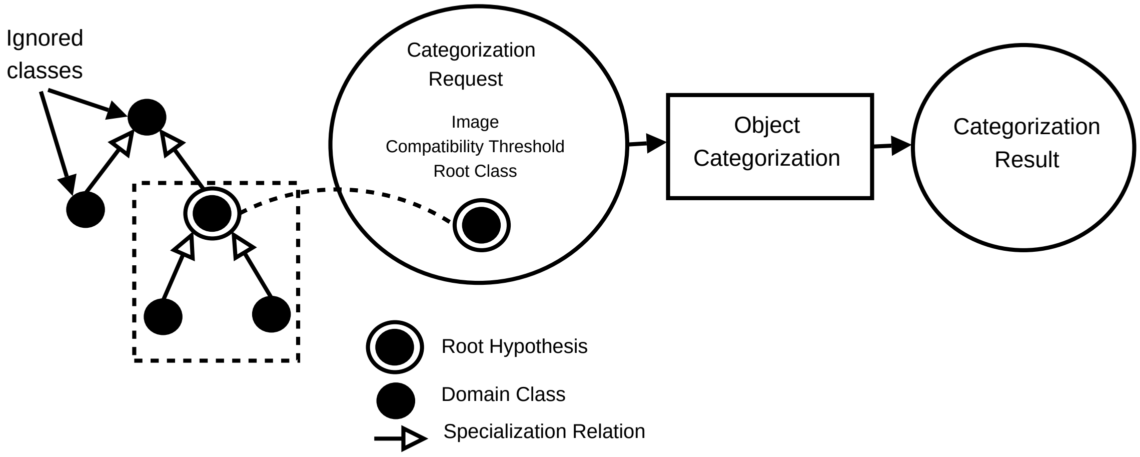


Figure 6.3: In this case, a specific class is used as a starting point for categorization. This implies that only three classes are taken into account during object categorization. If the categorization process selects a domain class, this implies that this class is a subclass of the root hypothesis.

- The $\mathcal{O}_{hypothesis}$ attribute, the value of this attribute is the class α used for generating the hypotheses to be verified in the image and then stored in $\mathcal{O}$.
- A compatibility degree ($\mathcal{O}_{compatibility}$) attribute. This degree is the degree of compatibility between α and between the evidence detected in the image.
- A set of visual attributes ($\mathcal{A}_{\mathcal{O}}$). For each attribute $a \in \mathcal{A}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}(a)$ is the set of the *detected* visual concepts. Each element of this set is a visual concept C associated with a confidence value $conf$: $(C, conf)$.
- A set of subpart attributes ($\mathcal{S}_{\mathcal{O}}$). For each subpart attribute $sp \in \mathcal{S}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}(a)$ is the set of *detected* subparts. Each element of this set is a *unknown object.*
- A set of spatial relation attributes ($\mathcal{SR}_{\mathcal{O}}$). For each spatial relation attribute $sr \in \mathcal{SR}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}(sr)$ is the set of *verified* spatial relations. Each element of this set is a *spatial relation* concept associated with a confidence value.

An example of domain class can be be found in table 6.1. Its associated unknown object is shown in table 6.2. Note that only visual attributes are involved in this example.

Unknown object initialization consists of using a domain class α so as to obtain an unknown object $\mathcal{O}$ such that:

- $\mathcal{O}_{hypothesis} = \alpha$.
- $\mathcal{O}_{compatibility} = 0.0$.
- $\mathcal{A}_{\mathcal{O}} = \mathcal{A}_{\alpha}$ and $\forall a \in \mathcal{A}_{\mathcal{O}},\ \mathcal{V}_{\mathcal{O}}(a) = \{\}$.

Domain Class	Sky
{	
SpatialAttributes :	
Position :	[Top]
ColorAttributes :	
Hue:	[Blue Grey]
Brightness:	[Light Dark]
TextureAttributes :	
Pattern:	[Smooth]
}	

Table 6.1: Visual description of domain class Sky.

- $\mathcal{S}_{\mathcal{O}} = \mathcal{S}_\alpha$ and $\forall sp \in \mathcal{S}_{\mathcal{O}},\ \mathcal{V}_{\mathcal{O}}(sp) = \{\}$.
- $\mathcal{SR}_{\mathcal{O}} = \mathcal{SR}_\alpha$ and $\forall sr \in \mathcal{SR}_{\mathcal{O}},\ \mathcal{V}_{\mathcal{O}}(sr) = \{\}$.

We have defined the *initializeUnknownObject* function which takes a domain class for parameter and returns an initialized unknown object. This function takes the domain class α for input parameter and returns the corresponding initialized visual object.

6.2.3 Hypothesis Generation

One key mechanism of categorization is *hypothesis generation*. A hypothesis is a set of visual concepts which have to be detected in a region for validating the hypothesis. Hypotheses are generated only in terms of visual concepts and not in terms of domain classes. One of our goals is to avoid the use of domain dependent low-level image processing algorithms. This means that low-level image processing mechanisms should not have direct access to high-level knowledge. The communication between low-level mechanisms and high-level mechanisms is enabled via the visual concept ontology which defines a shared vocabulary independent of the application domain.

A hypothesis is noted $\mathcal{H}$. For a domain class α, the associated generated hypothesis is defined as a set containing the attributes of α associated with their hypothetical values. The role of visual concept detection is to verify these hypotheses. A hypothesis $\mathcal{H}$ is produced by algorithm 5 from a class α.

For instance, the hypothesis associated with class *Sky* (table 6.1) is defined as:

$$\begin{gathered}\mathcal{H} = \{(Position, Top),\ (Hue, Blue),\ (Hue, Grey),\\ (Brightness, Light),\ (Brightness, Dark),\ (Pattern, Smooth)\}\end{gathered}$$

After hypothesis generation, hypothesis verification requires image segmentation, feature extraction and visual detection.

UnknownObject	$\mathcal{O}$
{	
$\mathcal{O}_{hypothesis}$:	Sky
$\mathcal{O}_{compatibility}$:	0.0
SpatialAttributes :	
Position :	[]
ColorAttributes :	
Hue:	[]
Brightness:	[]
TextureAttributes :	
Pattern:	[]
}	

Table 6.2: The unknown object associated with the domain class Sky is initially empty. This unknown object is filled with the visual concepts detected in the image. The compatibility the class is initially set to 0.0. In this example, $\mathcal{A}_{\mathcal{O}} = \{Position, Hue, Brightness, Pattern\}$.

Algorithm 5 *generateHypothesis*(α)

```
$\mathcal{H} \leftarrow \{\}$
for all $a \in \mathcal{A}_\alpha$ do
  for all $v \in \mathcal{V}_\alpha(a)$ do
    $\mathcal{H} \leftarrow \mathcal{H} \cup (a, v)$
  end for
end for
return $\mathcal{H}$
```

6.2.4 Image Segmentation

A mean shift segmentation algorithm [Comaniciu and Meer, 2002] is used for achieving the segmentation of the image contained in the categorization request. The mean shift segmentation algorithm belongs to the family of segmentation algorithms based on clustering techniques (e.g. by k-means). Mean shift clustering is a *non-parametric* density estimation technique based on kernel density estimation (i.e. Parzen Windows). Mean shift is an iterative procedure that shifts each data point to the average of data points in its neighborhood. Compared to k-means based segmentation (which creates clusters with hyperspherical boundaries), the robustness of this algorithm comes from the fact that the computation of the mean is restricted inside local windows. We use the C++ implementation available in the LTI-Lib [1]. The main parameters are the *size* of the mean-shift neighborhood and the maximum color difference between two regions that allow their merging. This algorithm has been selected for the following reasons: the number of pa-

[1] http://ltilib.sourceforge.net/

rameters to tune is low, the computing time is low as well, and its robustness is good.

The visual concepts contained in the generated hypothesis are used to drive automatic segmentation (see fig. 6.4).

One parameter of key importance for the selection of the regions resulting from segmentation is *the minimum area of the regions.* A region resulting from segmentation is selected if its area is greater than this minimum area.

Depending on the value of the size attribute in the generated hypothesis, two coefficients have been defined: $c_{LargeSize} \in [0,1]$ and $c_{SmallSize} \in [0,1]$ with $c_{SmallSize} < c_{LargeSize}$ (see table 6.3). The value of the minimum region area is obtained by multiplying the selected coefficient by the area of the image ($w \times h$).

Hypothesis	Minimum Region Area
$(Size, LargeSize)$	$c_{LargeSize} \times w \times h$
$(Size, AverageSize)$	$\frac{c_{LargeSize} - c_{SmallSize}}{2}$
$(Size, SmallSize)$	$c_{SmallSize} \times w \times h$

Table 6.3: A coefficient is associated with the different values of the size attribute. h and w are respectively the *height* and the *width* of the image to segment. Depending on the value of the size attribute in the generated hypothesis, different values are computed as the minimum region area required for selection.

Moreover, the value of the *Position* attribute in the hypothesis is used as a criteria for region selection. The position concepts provided by the visual concept ontology define 9 zones in the image. As seen in fig. 6.5, the different zones are defined by two coordinates (x_1, y_1) and (x_2, y_2). Let define (x_c, y_c) the center of gravity of the region R. The different criteria used for selecting or rejecting R are shown in table 6.4.

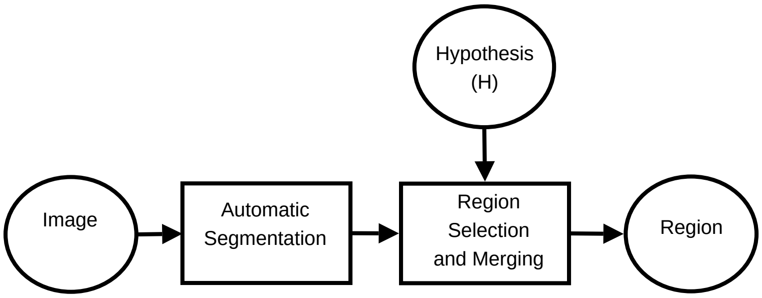


Figure 6.4: After automatic segmentation of the image, a region is obtained after *a selection and merging* process driven by a priori knowledge.

Knowledge contained in the generated hypothesis could be used for achieving better segmentation. Information related to the color, the texture and the shape of the expected objects could be used to drive segmentation. An approach based on the work presented in [Nazif and Levine, 1986] should be employed to make better use of knowledge provided

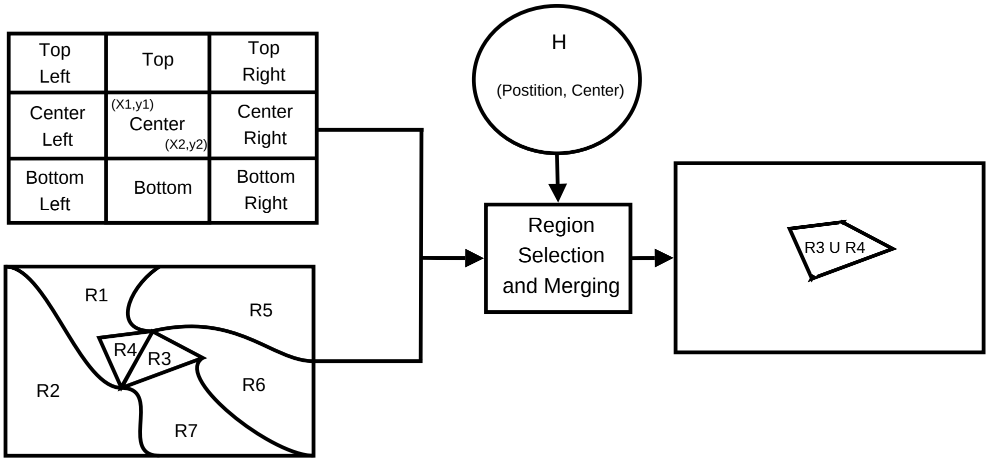


Figure 6.5: In this example, the hypothesis $(Position, Center)$ is contained in $\mathcal{H}$ and is used for *region selection and merging.* On the left, the position concepts provided by the visual concept ontology and a set of regions resulting from automatic segmentation are used as an input of the region selection and merging process. The regions R_3 and R_4 are selected as being in the center of the image. The final result of the segmentation process is the union $R_3 \cup R_4$.

by the generated hypothesis. This approach consists of using an expert system for guiding image segmentation. The proposed expert system is able to manage *focus of attention* rules. Such inference rules should be used for improving the region selection and merging flexible way (i.e. the rule base can be updated).

A *program supervision* [Clement and Thonnat, 1993] approach could also help to take into account the information contained in $\mathcal{H}$. In particular, visual concepts could be involved at the level of selection criteria. An example of selection rule could be: if the expected object has a granulated pattern then use a texture segmentation algorithm [Chen et al., 2004]. Different segmentation algorithms would then be used depending on the hypothesis.

6.2.5 Feature Extraction

We have seen that the result of the segmentation is a region R. As seen in fig. 6.1, feature extraction is driven by the visual concepts of the generated hypothesis. For each $(a, C) \in \mathcal{H}$, the feature extracted are defined by $\mathcal{F}(a)$.

The visual concept ontology defined in chapter 4 provides this knowledge. Depending of the category of visual concept to detect, different types of features are extracted from R. More details on low-level features associated with visual concepts can be found in section 4.4.

Hypothesis	Selection Criteria
$(Position,\ Top)$	$y_c < y_1\ and\ x_c \in [x_1, x_2]$
$(Position,\ Center)$	$y_c \in [y_1, y_2]\ and\ x_c \in [x_1, x_2]$
$(Position,\ Bottom)$	$x_c \in [x_1, x_2]\ and\ y_c > y_2$
$(Position,\ Top\ Left)$	$x_c < x_1\ and\ y_c < y_1$
$(Position,\ Center\ Left)$	$x_c < x_1\ and\ y_c \in [y_1, y_2]$
$(Position,\ Bottom\ Left)$	$x_c < x_1\ and\ y_c > y_2$
$(Position,\ Top\ Right)$	$x_c > x_2\ and\ y_c < y_1$
$(Position,\ Center\ Right)$	$x_c > x_2\ and\ y_c \in [y_1, y_2]$
$(Position,\ Bottom\ Right)$	$x_c > x_2\ and\ y_c > y_2$

Table 6.4: The *center of gravity* of the regions is used for their selection. (x_c, y_c) is the center of gravity of a region R candidate for selection. Given a hypothesis (first column), R is selected if the corresponding criteria (second column) is verified.

The role of the $featureExtraction$ function used in algorithm 4 is to extract a set of features from a region R. This function takes two arguments, a region and a set of features to extract (i.e. $\mathcal{F}(a)$). This function returns a feature vector $\mathbf{x} \in \mathbb{R}^n$.

For a visual concept of the category of texture pattern concepts (e.g $Granulated$, $Coarse$), a Gabor filter applied to R leads to a feature vector of dimension 128. Moreover, a feature vector of dimension 5 is obtained by the computation of grey-level co-occurence matrices. The total number of extracted features is 133.

6.2.6 Visual Concept Detection

The role of visual concept detection is to characterize region segmented by automatic segmentation in terms of visual concepts (e.g. $Blue$ for attribute Hue, $LargeElongation$ for attribute $Elongation$). The way visual concept detection is achieved is described in algorithm 6.

The $visualConceptDetection$ function has two input parameters. The first parameter is the visual concept C to be detected. The second parameter is a feature vector $\mathbf{x}$ obtained by feature extraction from a region R. The visual concept detection algorithm uses the visual concept detectors obtained during the visual concept learning phase described in chapter 5. The visual concept detector d_C is the detector associated with the visual concept C. $d_C(\mathbf{x}) \in [0, 1]$ measures the confidence degree $conf$ associated with the hypothesis "$\mathbf{x}$ is a representative sample of C". As explained in section 5.3, for a visual concept C, the associated visual concept detector is obtained by training a SVM with positive and negative samples of C. If this confidence degree is greater than a pre-defined threshold amb_{th}, then the visual concept C is considered as detected in the region R. The result of visual concept detection is a detected visual concept associated with a confidence degree.

Algorithm 7 presents another alternative visual concept detection algorithm. This

Algorithm 6 *visualConceptDetection*$(C, \mathbf{x})$

```
$conf \leftarrow d_C(\mathbf{x})$
$\{$C detected$\}$
if $conf \geq amb_{th}$ then
    $return(C, conf)$
end if
```

alternative takes into account the notions of ambiguity and distance reject. The distance and ambiguity reject notions are related to one of the research challenges of cognitive vision: *advancement of methods for continuous learning.* The use of these mechanisms for achieving object categorization enables to take into account previously unseen situations. Algorithm 7 shows that two sets are built by this alternative of the visual concept detection algorithm: a set of regions corresponding to the distance reject case and a set of regions corresponding to the ambiguity reject case.

For the *ambiguity reject case* (fig. 6.6), the set of rejected regions can be visualized and labeled by the expert. The resulting labeled regions can then be added to the set of annotated regions $\mathcal{AR}$ defined in chapter 5 and then used for visual concept learning purposes. Providing new annotated samples aims at reducing the number of ambiguous cases.

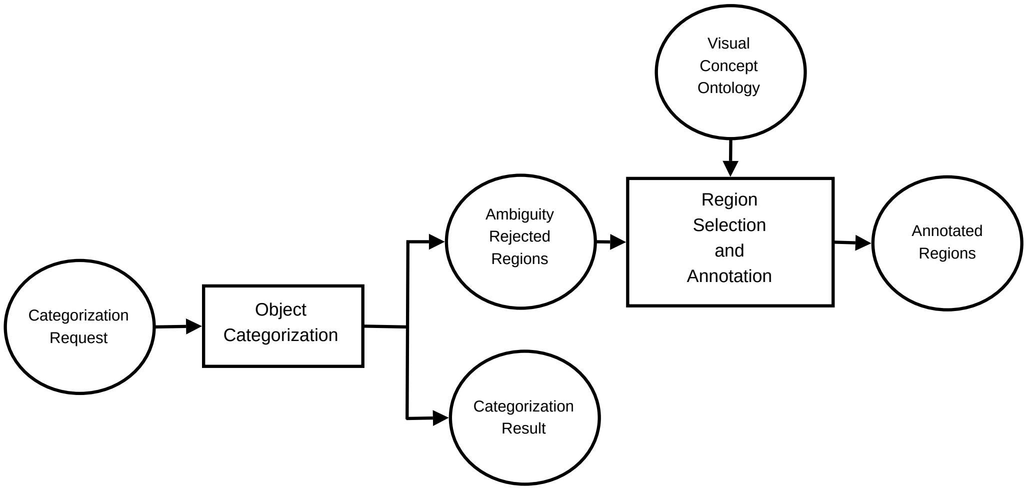


Figure 6.6: Illustration of the *ambiguity reject* case. The rejected regions can be labeled by the expert and then used for *visual concept learning* purposes.

For the *distance reject case* (fig. 6.7) , the rejected regions correspond to *unseen visual concepts.* For instance, for the attribute Hue, a region R is rejected if the feature vector $\mathbf{x}$ extracted from R is not detected by any of the visual concept detectors associated with the visual concept values of the attribute Hue. This means that the hue of R has not already been learned by the system.

Some rejected regions may be ignored. Other distance rejected regions may be labeled by visual concepts not already present in the visual concept ontology. The resulting set of annotated regions $\mathcal{AR}$ can then be used for visual concept learning purposes. In this case, the visual concept ontology has to be extended in order to take into account the new visual concepts used for annotation.

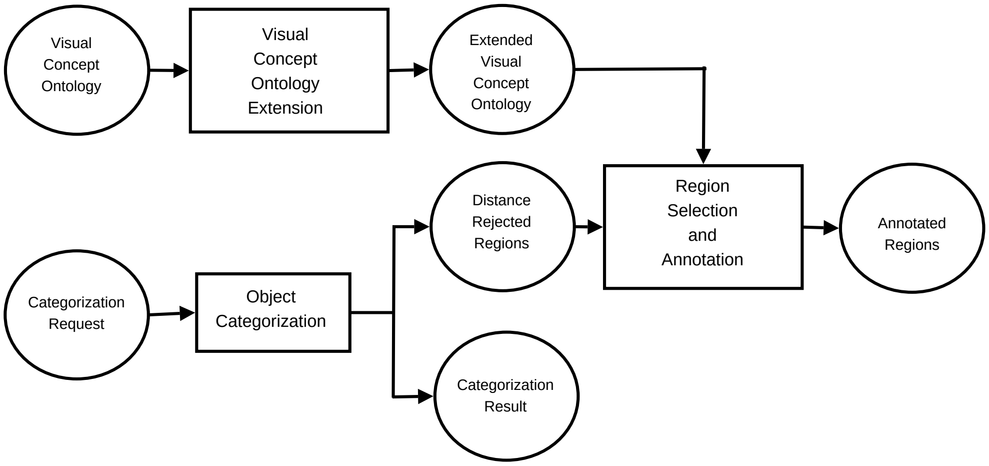

Visual Concept Ontology
Visual Concept Ontology Extension
Extended Visual Concept Ontology
Distance Rejected Regions
Region Selection and Annotation
Annotated Regions
Categorization Request
Object Categorization
Categorization Result

Figure 6.7: Illustration of the *distance reject* case. In this case, the distance rejected regions correspond to unseen visual concepts. Therefore, the visual concept ontology has to be extended with the new visual concepts. The regions can then be labeled by the expert and then used for *visual concept learning* purposes.

Regions rejected during categorization can be accumulated during several object categorization sessions corresponding to several categorization requests. Note that the expert is free to ignore the rejected regions. In this case, the categorization system can be based on algorithm 6. If the expert makes the effort to take into account the rejected region (i.e. by manual selection and annotation of the rejected regions), the performance of the system will be continuously improved.

Algorithm 7 $visualConceptDetection(C, \mathbf{x})$

```
$conf \leftarrow d_C(\mathbf{x})$
{$C$ detected}
if $conf \geq amb_{th}$ then
    $return(C, conf)$
end if
{Distance Reject}
if $\forall C_i \preceq_\theta a \mid a \in \mathcal{A}$ and $C \preceq_\theta a,\ d_{C_i}(\mathbf{x}) \leq dist_{th}$ then
    $distanceRejectedRegions \leftarrow distanceRejectedRegions \cup R$
end if
{Ambiguity Reject}
if $conf \in ]1 - amb_{th}, amb_{th}[$ then
    $ambiguityRejectedRegions \leftarrow ambiguityRejectedRegions \cup R$
end if
```

6.2.7 Unknown Object Filling

Unknown object filling uses the detected visual concepts. For a generated hypothesis $\mathcal{H}$, each detected visual concept associated with a confidence degree is stored in the unknown object. As shown in algorithm 8, the unknown object filling algorithm has three input parameters. The first parameter is a region R obtained by automatic segmentation. The second parameter is the generated hypothesis. The third parameter is the unknown object $\mathcal{O}$. This algorithm shows that when an attribute a is not already in the attributes of the unknown object $\mathcal{O}$, this new attribute is added to the set of attributes ($\mathcal{A}_{\mathcal{O}}$). Unknown object filling consists of accumulating the detected visual concepts in the values of the attributes of the unknown object. For that purpose, a feature extraction step is required before performing visual concept detection. Note that if a visual concept has already been detected and stored in the unknown object, feature extraction and visual concept detection are not performed.

Let consider the example of class *Sky* (table 6.1) and its associated unknown object (table 6.2). If the visual concept *Blue* is detected in a segmented region with a confidence value of 0.8, then the couple ($Blue$, 0.8) is appended to the set of values of the slot *Hue* of the visual object. The visual concept *Smooth* recognized with a confidence degree of 0.7 is stored in the values of the attribute *Pattern*. The visual concept *Top* recognized with a confidence degree of 1.0 is stored in the values of the attribute *Position*. The resulting filled unknown object is shown in table 6.5.

Algorithm 8 *unknownObjectFilling*$(R, \mathcal{H}, \mathcal{O})$

```
for all $(a, C) \in \mathcal{H}$ do
  if $a \notin \mathcal{A}_{\mathcal{O}}$ then
    $A_{\mathcal{O}} \leftarrow A_{\mathcal{O}} \cup a$
  end if
  if $(C, conf) \notin \mathcal{V}_{\mathcal{O}}(a)$ then
    $\mathbf{x} \leftarrow featureExtraction(R, \mathcal{F}(a))$
    $(C, conf) \leftarrow VisualConceptDetection(C, \mathbf{x})$
    $\mathcal{V}_{\mathcal{O}}(a) \leftarrow \mathcal{V}_{\mathcal{O}}(a) \cup (C, conf)$
  end if
end for
return $\mathcal{O}$
```

6.2.8 Recursion for Subparts

Let the class α be a class used for hypothesis generation during object categorization. The set of subpart attributes of α is $\mathcal{S}_{\alpha}$. Let $\mathcal{O}$ be the unknown object associated with α. As seen in algorithm 4, for each possible value of the subpart attributes $\mathcal{S}_{\alpha}$, the object categorization algorithm is recursively called. For each recursive call, one or several objects are returned. The unknown object $\mathcal{O}$ has a set of subpart attributes $\mathcal{S}_{\mathcal{O}}$. The subpart attributes of the unknown object are used to store the unknown objects returned by the

UNKNOWNOBJECT	$\mathcal{O}$
{	
$\mathcal{O}_{hypothesis}$:	Sky
$\mathcal{O}_{compatibility}$:	0.62
SPATIALATTRIBUTES :	
Position :	[(Top,1.0)]
COLORATTRIBUTES :	
Hue:	[(Blue,0.8)]
Brightness:	[]
TEXTUREATTRIBUTES :	
Pattern:	[(Smooth,0.70)]
}	

Table 6.5: The unknown object associated with the domain class Sky filled with the visual concepts detected in the image. In this example, no value has been assigned to the attribute *Brightness*. This means that neither the visual concept *Dark* nor the visual concept *Light* have been detected. The compatibility with the domain class sky is 0.62 $((1.0 + 0.8 + 0.70)/4)$. The importance of all attributes is 1.0. Since there is no subpart attribute in this example, only the visual compatibility is evaluated.

recursive calls.

For the segmentation of a subpart, the segmentation result obtained for the main class is reused. This means that subpart segmentation is performed only in the region resulting from the segmentation of the main class.

6.2.9 Unknown Object/Class Compatibility Evaluation

Once the unknown object $\mathcal{O}$ has been filled, the compatibility of $\mathcal{O}$ with the domain class $\mathcal{O}_{hypothesis}$ can be computed. The resulting degree of compatibility, $\mathcal{O}_{compatibility}$ is computed as a combination of the results of the evaluation of visual compatibility, subpart compatibility and spatial relation compatibility.

Visual Compatibility

Visual compatibility between an unknown object and a domain class is computed by algorithm 9. For all $a \in \mathcal{A}_{\alpha}$, a detected visual concept contained in $\mathcal{V}_{\mathcal{O}}(a)$ is selected if it is compatible with the possible values of a and if it has the highest associated confidence value compared to other compatible detected visual concepts. The overall visual compatibility is computed as a weighted sum taking into account the importance of the attributes ($w_{\alpha,a} \in [0,1]$) and the confidence associated with the selected visual concepts. The importance associated with each attribute is a priori defined by the expert during the knowledge acquisition phase (chapter 4).

Algorithm 9 $visualCompatibility(\mathcal{O}, \alpha)$

```
$visualCompatibility \leftarrow 0.0$
for all $a \in \mathcal{A}_\alpha$ do
  $max \leftarrow 0.0$
  {Select the compatible detected visual concept with the highest confidence}
  for all $v \in \mathcal{V}_\alpha(a)$ do
    for all $(C, conf) \in \mathcal{V}_O(a)$ do
      if $C = v$ and $conf > max$ then
        $max \leftarrow conf$
      end if
    end for
  end for
  $visualCompatibility \leftarrow visualCompatibility + w_{\alpha,a}.max$
end for
return $visualCompatibility / \sum_{a \in \mathcal{A}_\alpha} w_{\alpha,a}$
```

Subpart Compatibility

Subpart compatibility between an unknown object and a domain class is computed by algorithm 10. The computation of subpart compatibility is similar to the computation of visual compatibility. For all subpart attributes $sa \in \mathcal{S}_\alpha$, a detected subpart is selected in $\mathcal{V}_{\mathcal{O}}(a)$ if it is compatible with the possible values of sa (i.e. $\mathcal{V}(sa)$) and if it has the highest compatibility degree compared to other compatible detected subparts. The overall subpart compatibility is computed as a weighted sum taking into account the importance of the attributes (i.e. $w_{\alpha,sa}$) and the compatibility degree associated with the selected subparts. The importance associated with each subpart attribute is a priori defined by the expert during the knowledge acquisition phase (chapter 4).

Algorithm 10 $subpartCompatibility(\mathcal{O}, \alpha)$

```
$visualCompatibility \leftarrow 0.0$
for all $sa \in \mathcal{S}_\alpha$ do
  $max \leftarrow 0.0$
  {Select the compatible detected subpart with the highest confidence}
  for all $sp \in \mathcal{V}_\alpha(a)$ do
    for all $\mathcal{O}_{sp} \in \mathcal{V}_{\mathcal{O}}(a)$ do
      if $\mathcal{O}_{sp_{hypothesis}} = sp$ and $\mathcal{O}_{sp_{compatibility}} > max$ then
        $max \leftarrow O_{sp_{compatibility}}$
      end if
    end for
  end for
  $subpartCompatibility \leftarrow subpartCompatibility + w_{\alpha,sa}.max$
end for
return $subpartCompatibility / \sum_{sa \in \mathcal{S}_\alpha} w_{\alpha,sa}$
```

Spatial Relation Compatibility

Spatial relation compatibility between a domain class α and an unknown object $\mathcal{O}$ is evaluated by using the regions stored in the segmentation fact base. For a spatial relation attribute the domain class α $sra \in \mathcal{SR}_\alpha$, each spatial relation $Rel(\alpha, sa) \in \mathcal{V}_\alpha(sra)$ defines a possible relation between the domain class and the values of the subpart attribute sa. $Rel(\alpha, sa)$ is verified if there exists a region R associated with the unknown object $\mathcal{O}$ and a region R_{sp} associated with $\mathcal{O}_{sp}$ such that $\mathcal{O}_{sp} \in \mathcal{V}_\mathcal{O}(sa)$ and such that the spatial relation is verified at the pixel level by the $SpatialRelationVerification$ function. Inferential knowledge (e.g. symmetry, transitivity of spatial relations) can be used to infer new relations from the detected relations. The way spatial relation compatibility is computed is presented in algorithm 11. One limitation of this approach is that the *cardinality* of spatial relations is not taken into account.

Algorithm 11 $spatialRelationCompatibility(\mathcal{O}, \alpha, SegmentationFactBase)$

```
$spatialRelationCompatibility \leftarrow 0.0$
for all $sra \in \mathcal{SR}_\alpha$ do
  $max \leftarrow 0.0$
  for all $Rel(\alpha, sa) \in \mathcal{V}_\alpha(sra), sa \in \mathcal{S}_\alpha$ do
    for all $sp \in \mathcal{V}_\alpha(sa)$ do
      if $\exists((\mathcal{O}_{sp}, R_{sp}), (\mathcal{O},R)) \in SegmentationFactBase \mid \mathcal{O}_{sp_{hypothesis}} = sp \ and \ \mathcal{O}_{sp} \in \mathcal{V}_\mathcal{O}(sa) \ and \ SpatialRelationVerification(Rel, R_1, R_2) = true$ then
        $max \leftarrow 1.0$
        $\mathcal{SR}_\mathcal{O}(sra) \leftarrow \mathcal{SR}_\mathcal{O}(sra) \cup (Rel(\mathcal{O}, \mathcal{O}_{sp}), 1.0)$
        $\mathcal{V}_\mathcal{O}(sra) \leftarrow inferSpatialRelations(\mathcal{V}_\mathcal{O}(sra))$
      else
        $max \leftarrow 0.0$
      end if
    end for
  end for
  $spatialRelationCompatibility \leftarrow spatialRelationCompatibility + w_{\alpha,sra}.max$
end for
$return \ spatialRelationCompatibility / \sum_{sra \in \mathcal{SR}_\alpha} w_{\alpha,sra}$
```

6.2.10 Hypothesis Refinement

During object categorization, if a hypothesis is verified at a level of abstraction of the domain taxonomy then this hypothesis is refined by using the domain class hierarchy. Hypothesis refinement is based on a *depth-first traversal* of the domain taxonomy. More precisely, sub-classes of the current class are considered as hypotheses if the compatibility between the current class and the unknown object is greater than the threshold $comp_{th}$ defined in the categorization request.

This refinement process is illustrated in fig. 6.8. This figure is composed of two parts. On the top of the figure, an unknown object is filled with the visual concepts detected in the image to interpret.

In this figure, the resulting unknown object is compatible with the current class. Then, hypothesis refinement is performed in the bottom part of the figure. A new set of expected visual concepts is generated. A new set of detected visual concepts is then obtained by segmentation, feature extraction and visual concept detection. The unknown object is used to accumulate the visual concepts not previously detected. In case of non-compatibility, the current class and its subclasses are dropped.

If there is compatibility between the unknown object and the current class, the unknown object is placed in the categorization result. In this case, as seen in algorithm 4, the segmentation fact base is used to store the region of the input image which is compatible with the domain class. Storing segmented regions is particularly important for displaying the categorization results to the end-user.

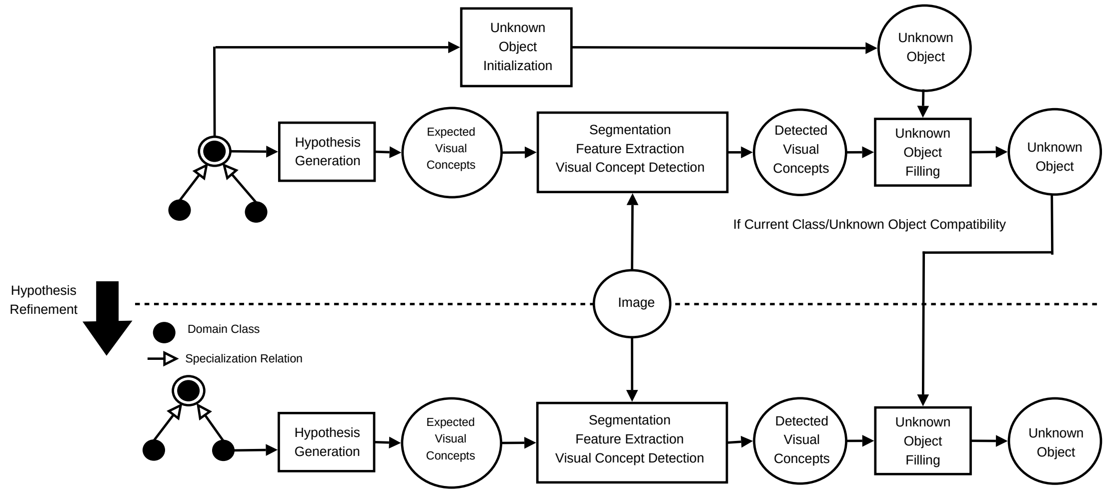


Figure 6.8: Illustration of hypothesis refinement. The unknown object is used to accumulate the detected visual concepts. In this bottom part of this example, the current class is a leaf. This implies that hypothesis refinement is not possible. Moreover, if the resulting unknown object is compatible with the current class, it is stored in the categorization result.

6.2.11 Categorization Result

The final categorization result is composed of the unknown object compatible with a class α. If the unknown object is not compatible with any class of the taxonomy, then the categorization result is empty. The resulting unknown object also contains subparts attributes ($\mathcal{S}_{\mathcal{O}}$) used to store unknown objects corresponding to the detected subparts.

Each region stored in the *segmentation fact base* is associated with one unknown object. This implies that the regions associated with the unknown objects contained in the categorization result be visualized by using the segmentation fact base. This is useful for displaying categorization results in a user-friendly way.

As seen in table 6.5, the categorization result is both in the terms of the domain (i.e. $\mathcal{O}_{hypothesis}$) and also in terms of visual concepts (i.e. the values of the attributes of the unknown object).

6.3 Conclusion

We have presented an *object categorization algorithm* which uses the knowledge acquired during knowledge acquisition and the visual concept detectors obtained by visual concept learning. The object categorization algorithm is initiated by a *categorization request*. This request contains an image, the root class from which categorization is started, and a compatibility threshold. The proposed categorization algorithm performs a depth-first traversal of domain taxonomy acquired during knowledge acquisition. Each class of the taxonomy is used for generating an hypothesis in terms of visual concepts to be detected in the image. Visual concept detection is achieved by image segmentation, feature extraction and by visual concept detection. Detected visual concepts are accumulated in an entity called an unknown object which stands as a fact base. This process is applied recursively for enabling subpart categorization. The compatibility between the unknown object and a class of the domain taxonomy is computed by evaluating the *visual compatibility*, the *subpart compatibility*, and the *spatial relation compatibility*. The performance of the system can be continuously improved by performing annotation of regions rejected during visual concept detection because of ambiguity or because of previously unseen visual concepts.

One limitation of the proposed categorization algorithm is that spatial relations are verified *a posteriori* after subpart detection. This point should be improved by taking into account the work presented in [Hudelot, 2005] where spatial relations are used to guide segmentation and to select regions resulting from segmentation.

Compared to appearance based techniques and geometric based techniques, the proposed approach really addresses the categorization problem at different levels of abstraction by considering a hierarchy of domain classes. Appearance based techniques and geometric based techniques often pose the object recognition problem as a flat classification problem. The acquired knowledge the complex problem of object categorization to be transformed into a set of well-defined visual concept detection problems. The combination of the different results of visual concept detection leads to a decision at the object level. The proposed approach keeps the semantic richness of knowledge-based approaches (e.g. specialization, subpart relations) by producing categorization results in terms of the domain and also in terms of visual concepts.

Chapter 7

Experimental Results

7.1 Introduction

In the previous chapters, we have described the different phases of our approach: *knowledge acquisition* (chapter 4), *visual concept learning* (chapter 5), and *object categorization* (chapter 6). Building an *operational object categorization system* implies sequentially going through these three phases.

The goal of the current chapter is to show how the proposed approach phases have been involved in three different tasks.

Section 7.2 illustrates how the proposed knowledge acquisition methodology has been used in the domain of palynology. This section shows how biological knowledge can be acquired and structured by using the formalism proposed in chapter 4. This knowledge acquisition phase has resulted in a taxonomy of pollen grains. The subparts of these pollen grains have also been taken into account. The visual description of the pollen grain has been driven by the visual concept ontology.

Section 7.3 shows how the visual concept learning and the visual concept detection approach have been applied to the problem of the learning and to the detection of texture visual concepts. This experimental part uses a set of highly varied natural texture patches: the *Brodtaz* texture set [Brodatz, 1966].

Section 7.4 presents how the complete approach has been used for *image indexing and retrieval* purposes. We have built an efficient domain specific semantic image indexing and retrieval system. This system has been obtained by applying the three phases: knowledge acquisition, visual concept learning and object categorization. We show how the categorization results produced by the categorization algorithm are used for indexing an image database. One strength of this indexing process is that it is symbolic and thus does not require storing cumbersome high-dimensional feature vectors. Semantic querying (e.g. query by keywords) is also enabled. Queries can be expressed in terms of the class of the object the user is looking for. The user can also integrate visual description information in the query. This chapter is concluded in section 7.5.

7.2 Knowledge Acquisition For Pollen Grain Visual Description

7.2.1 Introduction

This interest in the study of pollen grain comes from the A.S.T.H.M.A [1] European project. Palynology is the science that studies contemporary and fossil palynomorphs, including pollen and spores. The goal of this project was to achieve automatic pollen grain classification which is useful for clinicians to provide near real time accurate information on aeroallergens and air quality for sensitive users. The aim is to quantify the correlation between the environmental stress (so-called envi-contamination factor that is a combination of the concentration of allergens, the concentration of atmospheric pollutants including ozone and black dusts), and some indicators of the population health (medical data, hospitalization statistics, school and work absenteeism, medicine consumption). The task of the palynologist technician is to recognize the pollen particles present on a microscope slide, to give every pollen a name (family, genus, specie, group) and to finally produce a pollen spectrum for the given day. Not only because of the time required to obtain the pollen measurements from the sensor samples but also because possible human errors of counting and identifying the pollen grains can occur, it is of major interest to develop a system able to recognize the pollen grains and to count them per types. This means achieving automatic evaluation of the atmospheric pollen concentration. The capture of the pollen grains occurs by impact on a sticky surface that is then adequately prepared to be examined under the optical microscopy. Due to the complexity of the different types of pollen grains, palynologist knowledge is taken into account. The current section illustrates how the proposed knowledge acquisition approach is used as a support for the communication with the experts.

7.2.2 Pollen Grain Species

A pollen grain is usually an ellipsoidal three-dimensional object. A pollen grain is composed of two main parts: The peripheral layer called *exine* consists of a robust material which protects the grain. The inner part of the grain is called the *cytoplasm* which contains the genetic material. The size of a grain is usually between 20 and 60 microns.

Four pollen types have been selected in the A.S.T.H.M.A. project. The choice has been based on their frequencies in the study area as well as their allergenic capacities. The selection includes pollen Cupressaceae, Olea europea, Poaceae and Parietaria types. The pollen grain selected in this project are shown in fig. 7.1.

1. *Poaceae* pollen comprises a large number of species, most of them annual herbs, well distributed in the studied area. These pollens are one of the most important aeroallergens across Europe, although their relative contribution to pollinosis varies

[1] http://www-sop.inria.fr/orion/ASTHMA/asthma/asthma.html

regionally in relation to local vegetation, agriculture and climate. Although some species flower in winter, summer or autumn, most of them do it in spring. The pollen production per plant is, in general, scarce and it varies notably between species.

2. *Cupressaceae* comprises several species distributed in the Mediterranean area. Most of them are frequent in the cities as ornemental trees, blossoming during winter. Then Cupressaceae pollen is the main responsible of the winter pollinosis. The pollen grains are light and easily wind dispersed.

3. *Olea* pollen is represented by only one species, Olea europea. This species is distributed in the Mediterranean area and widely cultivated for oil production. Moreover, this pollen type is one of the main inducers of allergy. Pollen production per tree is very high and the blossoming occurs during the spring.

4. *Parietaria* pollen type involves some species of the Urticaceae family. Urtica and Parietaria species are ruderal, and abundant in urban environments. Parietaria pollen is a very important aeroallergen in the Mediterranean region. Its pollen is well dispersed due to its small size. Although most of the species flower in the spring season, some Parietaria do it in autumn.

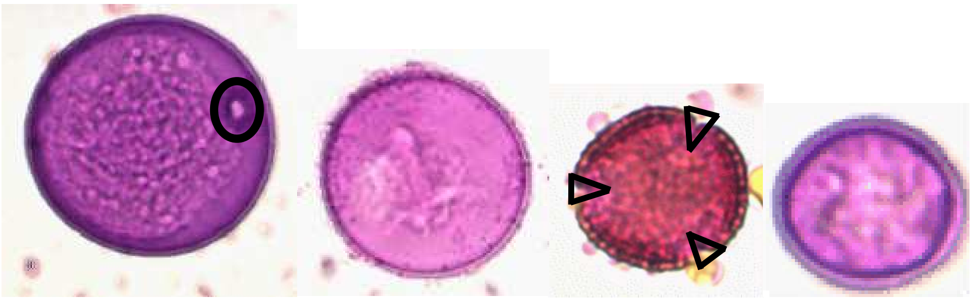


Figure 7.1: The four types of pollen grains used in this knowledge acquisition experiment (sorted by *size* from the *left* to the *right*): *Poaceae*, *Cupressaceae*, *Olea*, *Parietaria*. Two types of subparts are highlighted. The Pori of the Poaceae pollen grain (Ellipse) and the colpi of the Olea pollen grain (Triangles).

7.2.3 Knowledge Acquisition

Pollen Grain Taxonomy/Partonomy

The knowledge acquisition phase involving experts has led to the taxonomy/partonomy of domain classes illustrated in fig. 7.2. Fifteen domain classes have resulted from this knowledge acquisition process. The most discriminative subparts have been taken into account.

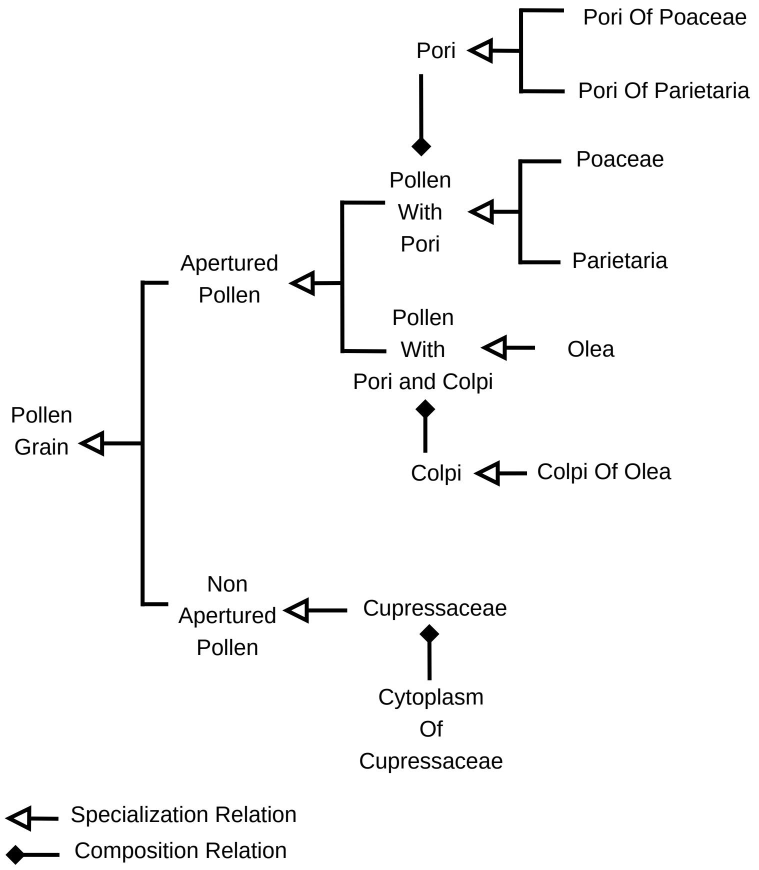


Figure 7.2: Taxonomy/Partonomy of the most allergenic pollen grains.

Pollen Grain Visual Description

Table 7.1 shows how the visual concepts are used for describing the classes presented in fig. 7.2. The visual concepts used for pollen grain description have been obtained during a knowledge acquisition process. 17 different visual concepts are used in this knowledge base. Two different topological relations between classes and their subparts are also used (i.e. $TPP(X,T)$, $NTTP(X,Y)$) . 18 domain classes are involved in the knowledge base. 8 subpart relations are used. A detailed example for the pollen grain *Olea* and its subpart *Colpi* can be found in tables 7.2 and 7.3. The visual concept ontology is playing a role of interface between the palynologists and the computer vision experts. It gives a user-friendly access to low-level features and algorithms (see section 4.4). For instance, color coherence vectors for characterizing the color of the objects of interest and the Gabor features for characterizing the texture of the objects of interest.

Context Description

The acquired knowledge describes the appearance of the pollen grains in given acquisition context. This context is defined in table 7.4. The acquisition sensor used was a CCD

Expert terminology	Spatial Attribute	Color Attributes	Texture Attributes	Subpart Of	superclass
Pollen	-	-	-	-	-
Non Apertured Pollen	-	-	-	-	*Pollen*
Apertured Pollen	-	-	-	-	*Pollen*
Pollen with Pori	-	-	-	-	*Apertured Pollen*
Pollen with Colpi	-	-	-	-	*Apertured Pollen*
Pollen with Por(e) and Colpi	-	-	-	-	*Aperured Pollen*
Cupressaceae	**Circular Surface and Average Size**	**Brilliant Blue**	**Slightly Granulated**		*Non Apertured Pollen*
Poaceae	**Elliptical or Circular Surface Small or Average or Large Size**	**Dark Pink**	**Granulated Texture**	-	*Pollen with Pori*
Parietaria	**Elliptical Surface and Small Size**	**Brilliant**	**Smooth Texture or Granulated Texture**	-	*Pollen with Pori*
Olea	**Elliptical or Circular Surface Average Size**	**Dark Red**	**Not Regular**	-	*Pollen with Pori and Copli*
Aperture	-	-	-	*Apertured Pollen*	**Subpart**
Pori	-	-	-	*Pollen with Pori or Pollen with Pori and Colpi*	*Aperture*
Colpi	-	-	-	*Pollen with Colpi or Pollen with Pori and Colpi*	*Aperture*
Exine	-	-	-	*Pollen*	**Subpart**
Cytoplasm	-	-	-	*Pollen*	**Subpart**
Pori of Poaceae	**Elliptical Surface**	**Very Light**	**Smooth Texture**	*Poaceae*	*Pori*
Colpi of Olea	**Triangular Surface and Small Size**	**Very Light**	**Smooth Texture**	*Olea*	*Colpi*
Pori of Parietaria	**Elliptical Surface and Small Size**	**Very Light**	**Smooth Texture**	*Parietaria*	*Pori*
Cytoplasm of Cupressaceae	**Polygonal Surface and Large Size**	**Dark**	**Not Regular**	*Cupressaceae*	*Cytoplasm*

Table 7.1: The four most allergenic pollen grains (and their subparts) described with visual concepts. Concepts provided by the visual concept ontology are in boldface. Domain knowledge is in italic. For all the domain classes, the *Position* attribute is set to *Center*.

Class	*Olea*
SubParts:	
Colpi	[*Colpi Of Olea*]
SuperClass:	*Pollen With Pori And Colpi*
SpatialAttributes :	
Geometry :	[**EllipticalSurface CircularSurface**]
Position :	[**Center**]
Size :	[**AverageSize**]
ColorAttributes :	
Hue:	[**Red**]
Brightness:	[**Dark**]
TextureAttributes :	
Repartition:	[**Not Regular**]
SpatialRelations :	
SpatialRelation r1:	[**TTP**(*Olea*,*Colpi*)]

Table 7.2: High level description of domain class *Olea*.

Class	*Colpi Of Olea*
SuperClass:	*Colpi*
SpatialAttributes :	
Geometry :	[**TriangularSurface**]
Size :	[**SmallSize**]
ColorAttributes :	
Brightness:	[**Very Light**]
TextureAttributes :	
Repartition:	[**Not Regular**]

Table 7.3: High level description of the subpart *Colpi Of Olea*.

camera combined with a light microscope. The magnification of the microscope was 60. An artificial dye was used for better visualization of the pollen grains.

AcquisitionContext	*Pollen Grain Acquisition Context*
Sensor	[**CCD Camera with Light Microscope**]
Magnification	[60]
Dye	[Fuchsin]

Table 7.4: Acquisition context during the A.S.T.H.M.A project.

7.2.4 Conclusion

The proposed knowledge acquisition methodology has enabled the acquisition of the knowledge related to the visual description of four types of the most allergenic pollen grains. The most discriminative attributes of the four pollen grains are: *Geometry*, *Size*, *Hue*, *Brightness*, *Pattern*, and *Repartition*. The subparts and the spatial relations between the main objects and these subparts are also discriminative elements. The proposed knowledge acquisition methodology enables to structure expert knowledge in two parts: the *taxonomy/partonomy of domain classes* and the *visual concept based description*. The first part is independent of any computer vision application and can be reused for other purposes (e.g. knowledge sharing and management). The second part of the acquired knowledge is dependent of the acquisition conditions of the images of interest.

7.3 Visual Concept Learning and Detection : Application to Texture Patches

7.3.1 Introduction

The previous section has shown how the proposed knowledge acquisition phase has been used for acquiring knowledge in the domain of palynology. The current section is not dedicated to a specific domain of interest but is rather focused on visual concept learning and detection. The goal of this section is to show that a varied set of texture visual concepts can be learnt and detected. The Brodtaz texture set [Brodatz, 1966] has been used for this experiment. This means that *no* domain knowledge is involved in this experiment. The visual concept detectors obtained during a visual concept learning phase are used for visual concept detection purposes.

A cognitive experiment has been performed in [Rao and Lohse, 1993]: a subset of 56 Brodatz texture images has been given to 20 persons who were asked to group the images into classes. The clusters were formed by evaluating the following symbols (between 1 and 9): contrast, repetitiveness, granularity, randomness, roughness, density, directionality, complexity, coarseness, regularity, orientation. The authors of this experiment have then chosen labels associated with each cluster which have been used to build the texture concept hierarchy of the visual concept ontology. Our goal is to see if this way of naming and understanding texture images can be reproduced artificially by a visual concept learning and then by visual concept detection.

7.3.2 Visual Concept Learning

Image Sample Annotation

All texture patches are of dimension 512x512 and have been split in 16 pieces so as to obtain images of size 128x128. To perform the learning process described in this section, we have used texture visual concepts to describe Brodatz images. As seen in table 7.5, each Brodatz image is representative of only one texture visual concept. This implies that *no segmentation* has been required during the annotation. The annotation process has led to a set of annotated regions $\mathcal{AR}$ (see section 5.2). The total number of *different* visual concepts used for annotation is 10: *Repetitive*, *Random*, *Regular*, *Oriented*, $Uniform$, *Directional*, *Granulated*, Not *Granulated*, Not *Repetitive*, Not *Random*. The total number of patches described by visual concepts is 896 (56x16). All the texture patches have been manually gathered in clusters and then annotated by the visual concepts used for describing their associated clusters. For instance, the region sample ($R26$) of cluster G (table 7.5) leads to the following annotations (i.e. a subset of $\mathcal{AR}$): $\{(R26, \mathit{Repetitive})$, $(R26, \mathit{Oriented})$, and $(R26, Uniform)\}$.

Texture feature extraction algorithms (i.e. Gabor filters (128 features) and cooccurence matrices (5 features computed for 8 different directions) have been applied to all 128x128 image to obtain a training set. Feature selection has reduced the number of features from 168 to 20 by Linear Discriminant Analysis (LDA).

The visual concept learning algorithm (algorithm 1) described in section 5.3 has been used to produce a set of texture concept detectors D (each texture concept detector is a function from $\mathbb{R}^{20}$ to $[0, 1]$):

$$D = \{d_{\mathit{Repetitive}}, d_{\mathit{Random}}, d_{\mathit{Regular}}, d_{\mathit{Oriented}}, d_{Uniform}, d_{\mathit{Directional}}, d_{\mathit{Granulated}},$$
$$d_{Not\ \mathit{Granulated}}, d_{Not\ \mathit{Repetitive}}, d_{Not\ \mathit{Random}})$$

7.3.3 Visual Concept Detection

Due to the fact that the size of the training set is small, we have evaluated the visual concept detection by *N-fold cross-validation* (N=56). This evaluation approach consists of dividing the training set in N subsets. Then, visual concept learning, and visual concept detection are repeated N times. Feature extraction has been performed only once for all texture patches. At each step, a subset is selected and used for obtaining visual concept detection results. The remaining N-1 subsets are used for visual concept learning. Results presented in table 7.6 are the average of the N classification results obtained by using the

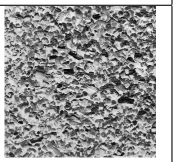
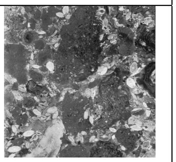
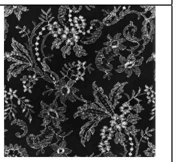
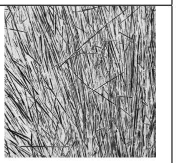
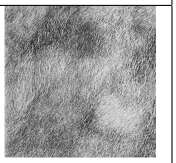
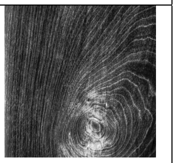
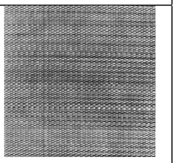
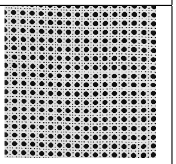
Cluster	Annotations	Sample
A	Granulated	
B	Random, Not Granulated, Not Repetitive	
C	Not Random, Not Repetitive, Not Directional	
D	Random, Repetitive	
E	Random	
F	Directional	
G	Repetitive, Oriented, Uniform	
H	Directional	

Table 7.5: Clusters obtained by an experiment described in [Rao and Lohse, 1993]. The second columns contains the visual concepts used for describing the clusters. Any texture patch belonging to a cluster is annotated by the corresponding visual concepts.

visual concept detection algorithm 6 presented in section 6.2.6. The threshold amb_{th} was set to 0.75.

If a visual concept C is detected in a texture patch not annotated by C, this is a *false positive* (see fig. 7.3). If a visual concept C is not detected in a texture patch annotated by C, this is a *false negative*. An example of false positive is given in fig. 7.3. In this example, the confidence value associated with the visual concept *Granulated* is greater than the threshold amb_{th} and is thus considered as detected. This texture patch has *not* been annotated with the *Granulated* visual concept. This means that this detection corresponds to a false positive.

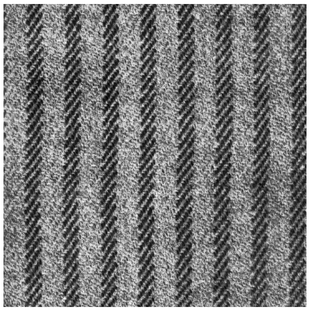


Figure 7.3: Brodatz texture sample. This sample is annotated with the following visual concepts: *Repetitive*, *Oriented*, $Uniform$. Visual concept detection on this path results in a set of detected visual concepts with a confidence value For instance, (Repetitive,0.8),(Oriented,0.79), (Uniform,0.85) and (Granulated, 0.76). In this case, the couple (Granulated, 0.76) is a false positive.

Concept	False Positive	False Negative	True Positive
Repetitive	25.3%	6%	94%
Random	17.8%	24.1%	75.9%
Regular	23.9%	13.1%	86.9%
Oriented	7.8%	2.2%	97.8%
Uniform	23.3%	1.2%	98.8%
Directional	19.9%	27.3%	72.7%
Granulated	8.8%	0%	100%
Not Granulated	22.1%	29.1%	70.9%
Not Repetitive	21.2%	36.9%	63.1%
Not Random	12.7%	26.5%	73.5%

Table 7.6: Texture concepts detection results.

7.3.4 Conclusion

Based on cognitive experiment involving humans, a set of texture patches have been annotated by visual concepts. We have used the results of this experiment for achieving the learning of a set of texture concepts. An evaluation has then been performed to see if the way humans do texture concept detection could be reproduced. The obtained results show that high true positive rate are obtained (especially for the visual concepts : *Repetitive*, *Oriented*, $Uniform$, *Granulated* for which the true positive rate is greater than 94%). The obtained false positive and false negative rates are not very low. One explanation for that is the low number of samples used during visual concept learning. Providing more samples would probably reduce both the false positive rate and the false negative rate.

7.4 Semantic Image Indexing and Retrieval

7.4.1 Introduction

One key issue in the design of image indexing and retrieval systems is their degree of user-friendliness. This is related the *communication* functionality of cognitive vision systems. As explained in [Boujemaa and Fauqueur, 2003], the query by visual example paradigm does not address all the end-user needs. In this case, retrieval results express a global or local (e.g. [Carson et al., 1999]) visual similarity. The query by visual example paradigm only entails the use of low-level visual information. As a consequence, high-level semantics is ignored.

Our goal is to find query mechanisms for image search similar to query mechanisms for text search. This kind of querying is now commonly used (e.g. the Google search engine) and has proved its efficiency. For reaching this goal, we propose to use our object categorization approach for achieving semantic image indexing. We are going to see that this indexing scheme is symbolic and does not require storing image regions or feature vectors. The symbolic nature of the indexing also enables straightforward symbolic and text-based querying of a set of indexed images.

This section is structured as following: first, we give an overview of how the problem of signal/semantics integration is tackled in image retrieval community. Then, we detail all the phases (i.e. knowledge acquisition, visual concept learning, object categorization for indexing, and retrieval) that has led to an efficient semantic image indexing and retrieval system dedicated to the domain of transport vehicles. After a few remarks on implementation issues in section 7.4.7, a conclusion on the interest of our approach for image indexing and retrieval purposes can be found in section 7.4.8.

7.4.2 State of The Art

The image retrieval community is currently looking for a framework for signal/semantic integration which brings user-friendly interaction and that produces good results. These two points are necessary conditions for obtaining a successful image indexing and retrieval system.

Image conceptual indexing and retrieval paradigm is now a topic of great interest in the image retrieval community. This stems from the limits of the query by example paradigm where image samples have to be provided : as explained in [Town and Sinclair, 2004], one or several query image(s) cannot capture the conceptual essence of the user query. Moreover, the user-friendliness of image retrieval systems is of key importance. Querying databases by manipulating high-level concepts (especially in a textual form) is something very natural for the end-user. Image conceptual indexing and retrieval can be achieved by different approaches.

Some techniques use manual annotations of images [Soo et al., 2003]. In this case, retrieval uses these annotations. Image processing is not used for indexing and retrieval.

In [Belkhatir et al., 2004], a multi-faceted framework based on conceptual graphs (see section 2.3.2) is proposed. A facet is called the *visual semantics facet* and describes the image high-level semantic content. Another facet is called the *object facet*. This facet brings the structure to take into account the visual entities within an image. A third facet is called the *signal facet* and describes the image signal content in terms of symbolic perceptive features and consists of characterizing the visual entities in the image with signal concepts. Two categories of signals concepts are available: *color* and *texture* concepts. This approach makes a clear distinction between the three levels of abstraction involved in the image interpretation problem. Moreover, the conceptual graph formalism brings a lot of expressiveness for the representation of high-level knowledge. Thanks to the conceptual graph formalism, the correspondence between query graphs and indexed images can be computed efficiently and in a user-friendly manner. The geometry and the size of the visual entities composing the image could be easily integrated in this formalism.

The notion of *visual thesaurus* is presented in [Picard, 1995] as a tool which group visual similarities (e.g. texture similarity) like a text thesaurus helps group semantically similar words. A visual thesaurus is not dependent of a specific application domain. The authors of this work also explain that common sense knowledge can emerge by learning the associations between the elements of the visual thesaurus.

In [Fauqueur and Boujemaa, 2003], querying is based on a logical composition of region templates. The authors of this work propose a retrieval paradigm based on the user mental image. This approach relies on the unsupervised generation of a *visual thesaurus* from which query by logical representation of regions can be performed. User semantics emerges from the combination of regions. One goal of the authors of this work is to reach a higher semantic level.

In [Li and Wang, 2003], a statistical approach learns keywords describing image regions. A set of manually annotated images is used to enable learning. Such systems are able to produce a symbolic explanation of a new image. Due to the fact that no a priori knowledge is used, this approach often lead to semantically inconsistent image annotation.

As explained by the authors of [Li and Wang, 2003], a rule-based engine should be used to improve image interpretation consistency.

In [Mezaris et al., 2004], querying is based on an object ontology which defines the mapping between low level descriptors and intermediate level semantic notions. The system is used in two phases. Each concept (color, position, size, shape) of the proposed ontology is defined by the appropriate range of numerical values of the corresponding low level descriptors computed in image regions (e.g. luminance, hue). These generic constraints lead to coarse retrieval results. User feedback is then used to train support vector machines dedicated to constraint refinement. This approach relies on a cumbersome numerical descriptor database and does not propose a well defined formalism for high-level knowledge. Moreover, this work does not really address the issues related to the acquisition and the formalization of high-level semantic categories.

In [Town and Sinclair, 2004], an image retrieval approach based on an extensible ontology is proposed. Querying is achieved by combining ontological concepts (e.g. size, location, color, semantic category). This combination is constrained by a grammar. Mapping between image data and concepts is based on supervised machine learning techniques (i.e. multi-layer perceptrons and radial basis networks).

A look on the state of the art shows that the image retrieval community is trying to find a trade off between the amount of work needed to build image indexing and retrieval systems (e.g. supervised learning, manual annotation) and semantic richness.

7.4.3 Knowledge Acquisition

We have seen that one important aspect of *image indexing and retrieval* systems is their user-friendliness. This can be achieved by enabling the end-user to have access to the object classes of his/her domain of interest. The first phase of our approach, *knowledge acquisition*, is used for acquiring this knowledge.

In the following sections, we are interested in the *domain of transport vehicles* (e.g. cars, motorbikes, aircrafts). Image samples containing such objects can be found in table 7.8. The visual description of all object classes of interest is given in table 7.7. The following discriminative attributes of $\mathcal{A}$ are used : Hue, *Brightness*, *Geometry*, *Position*, *Pattern*. The corresponding taxonomy/partonomy is shown in fig. 7.4.

Note that three visual concepts have been added to the visual concept ontology as sub-concepts of $PolygonalSurface$. $AircraftGeometry$, $CarGeometry$, *MotorbikeGeometry*. If this specialization is not made, the visual concept based description of the three classes: $Aircraft$, Car, and *Motobike* is not discrimative. These new concepts correspond to the generalized and schematized image that human beings have of the shape of these three object classes.

Class	*Hue*	*Brightness*	*Geometry*	*Position*	*Pattern*
Aircraft			*AircraftGeometry*	*Center*	
Car			*CarGeometry*	*Center*	
Motorbike			*MotorbikeGeometry*	*Center*	
Sky	*Blue Grey*	*Dark Light*		*Top*	*Smooth*
Tarmac	*Grey Black*			*Bottom*	*Uniform*
Grass	*Green*			*Bottom*	*Uniform*

Table 7.7: Visual description of the objects classes of the domain of *transport vehicles*. All these visual concepts are used for visual concept learning purposes and then for object categorization purposes. In addition, the visual concept *StrongElongation* is used as a possible value of the attribute *Elongation* for the class Aircraft.

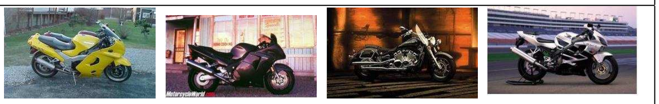
Image Category	Sample images
Car	
Motorbike	
Aircraft	
Background	

Table 7.8: Sample images associated with each class of objects of interest. Background images also contain complex objects.

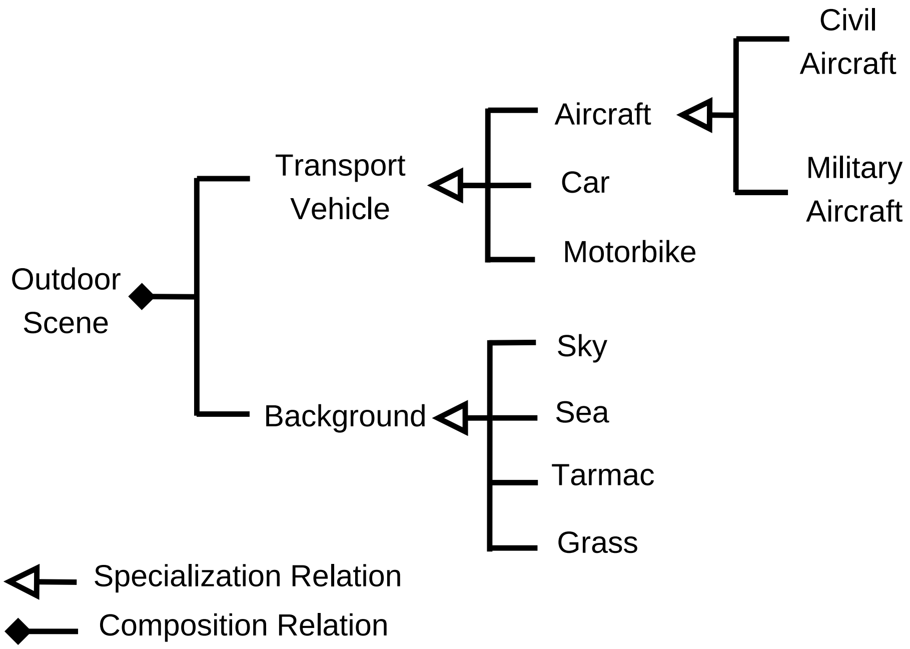


Figure 7.4: Structure of the knowledge base used in this experiment. Both specialization and composition relations are defined in the knowledge base. This knowledge is independent of any computer vision layer.

7.4.4 Visual Concept Learning

For achieving visual concept learning, we have used the image training set which is structured as described in tables 7.9 and 7.8. The image database freely available online [2] was used to apply our methodology. Four categories of images in this database are taken into account: Aircraft, Car, Motorbike and Background. This image database fits well with the hypotheses of our work: they contain one object of interest and the acquisition conditions are uniform (the size of the objects of interest does not vary too much from an image to another). A *background image* (see table 7.8) is defined as *not containing* any object of interest.

Image Category	Number of Images
Aircraft	400
Car	250
Motorbike	200
Background	400

Table 7.9: A total number of 1250 images is contained in the image training set.

We have used a combination of the three approaches described in section 5.2: *manual*

[2] http://www.vision.caltech.edu/feifeili/Datasets.htm

segmentation, 3-D model-based approach, and the weakly-supervised approach.

The weakly-supervised approach is particularly well adapted for obtaining samples of uniform and/or smooth regions (e.g. Sky, Grass, Tarmac). A 3-D model of an aircraft has also been used for generating 20 additional region samples. Manual segmentation has also been used to obtain the samples of complex objects which are difficult to automatically segment or for which no 3-D models was available (Cars and Motorbikes).

Figure 7.5 shows some regions of a cluster resulting from a weakly-supervised visual concept learning process for $a = Hue$. In this case, the resulting cluster is annotated by the visual concept $Green$. Figure 7.6 shows some regions of a cluster resulting from weakly-supervised visual concept learning process for $a = Elongation$. In this case, the resulting cluster is annotated by the visual concept $StrongElongation$. In these two examples, the number of clusters has initially been set to 15. A k-means algorithm is used. The result of visual concept learning is a set of 12 visual concept detectors:

$$D = \{d_{Blue}, d_{Grey}, d_{Green}, d_{Black}, d_{Dark}, d_{Light}, d_{AircraftGeometry}, d_{CarGeometry},$$
$$d_{MotorbikeGeometry}, d_{StrongElongation}, d_{Smooth}, d_{Uniform}\}$$

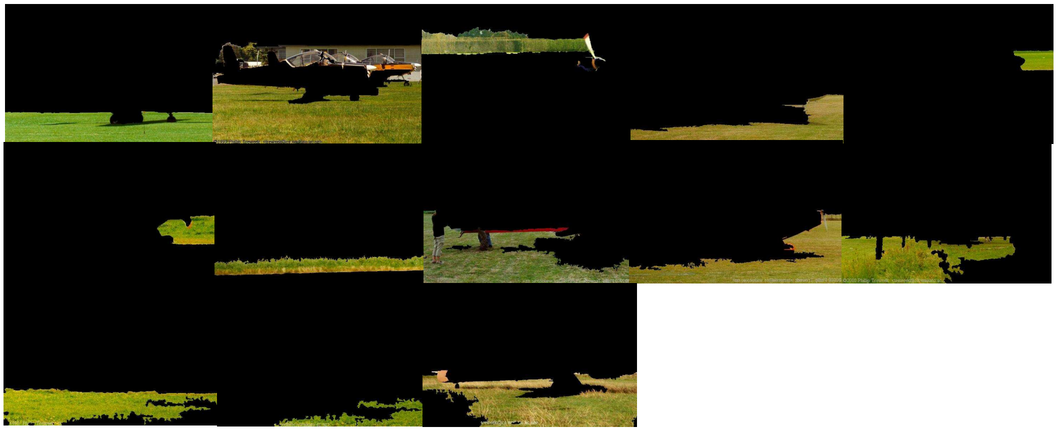


Figure 7.5: A set of regions obtained by a weakly-supervised approach ($a = Hue$). This cluster is labeled with the visual concept $Green$. All these regions result from automatic segmentation of the image training set.

7.4.5 Object Categorization for Indexing

The semantic image indexing process relies on the object algorithm presented in chapter 6. As seen in fig. 7.7, the indexing of an image database of N images implies sending N object categorization requests containing, for each image:

- the image to index
- the compatibility threshold ($comp_{th} = 0.8$).

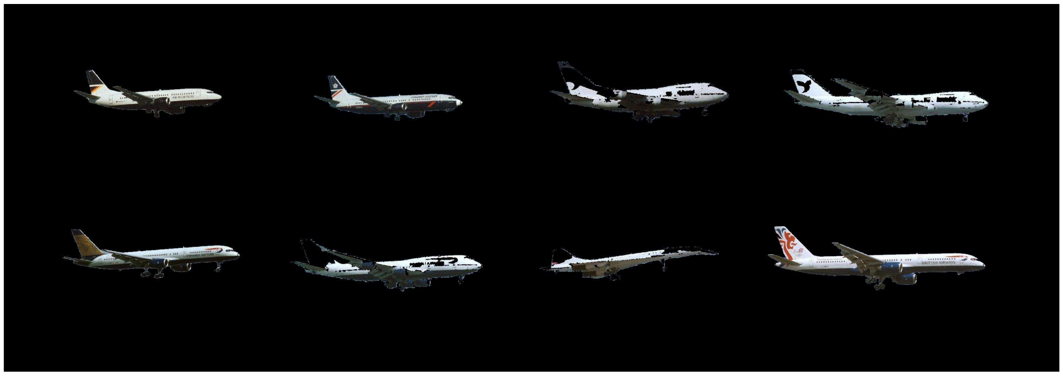


Figure 7.6: A set of regions obtained by a weakly-supervised approach ($a = Elongation$). This cluster is labeled with the visual concept *StrongElongation*. The good quality of the automatic segmentation is explained by the fact that the background of the associated non-segmented images is a clear sky.

- the root class (i.e. *OutdoorScene*) used as the starting hypothesis of the object categorization process.

As explained in chapter 6, the output of object categorization is a categorization result containing an *object* compatible with a class of the domain class hierarchy. The resulting categorized object can be stored in a linear form as a set of pairs attribute/value. This linear form enables efficient storing in a *relational database* and is composed of the following elements:

1. An *identifier* associated with the detected object.
2. The *class* of the detected object associated with a compatibility value.
3. The *identifier* of the main object of which the detected object is a subpart of.
4. The detected values of the visual attributes ($\mathcal{A}$).
5. The detected values of spatial relations ($\mathcal{SR}$).
6. The *path* to the indexed image.

Here is the example of the indexing of an image representing an *OutdoorScene* and composed of *Sky* and of *Grass*:

```
Id=1023;Class=(OutdoorScene,0.86);path=~/imageDatabase/img1082.jpg
Id=1024;Class=(Sky, 0.81); MainObject=1023;Hue=(Blue, 0.80);Brightness=(Light, 0.90);
...Pattern=(Smooth, 0.75)
Id=1025;Class=(Grass, 0.92); MainObject=1023;Hue=(Green, 0.95);Pattern=(Uniform, 0.90)
Id=1026;Class=(Aircraft, 0.85); MainObject=1023; Geometry=(AircraftGeometry, 0.80);
...Elongation=(StrongElongation, 0.90)
```

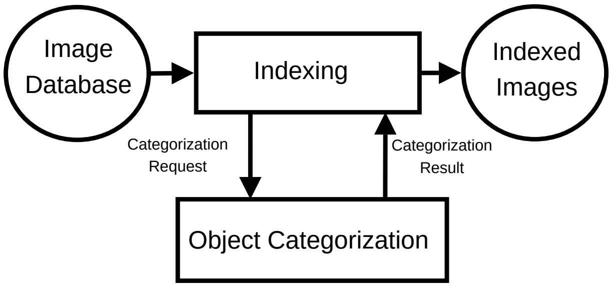


Figure 7.7: The indexing of an image database is based on the object categorization process described in chapter 6. The categorization result computed for one image of the image database is used for indexing this image. Object Categorization is applied to all the images of the input image database.

From this structure, *querying* can be achieved as a *logical combination* of domain classes and of visual concepts. This is equivalent to a problem of querying a relational database. This can be done very efficiently. Here is an example of a query:

```
(class=Sky and Brightness=Light) or class=Aircraft or class=Motobike
```

The indexing time for one image of resolution 600 × 400 is about 1 second on a PC (running the GNU/Linux operating system) with the following characteristics: Intel Pentium 4 CPU at 3.06GHz, 1.5Gb of Ram. The set of images to index is structured as following : 500 aircraft images, 500 motorbike images, 250 car images and 600 background images. A total number of 1850 images has been indexed. No image used for visual concept learning has been involved in the indexing process.

Segmentation

Examples of *automatically* segmented images are shown in table 7.10. The two first rows of this table show satisfying segmentation results. The third row of this figure shows a case of segmentation failure.

Feature Extraction

Given the visual description in terms of visual concepts of the object classes of interest (see table 7.7), the following low-level features are used: color histograms, color coherence vectors, geometric features (e.g. elongation, area), Sift features, Gabor features and co-occurence matrices.

An example of a set of detected key-points is shown in fig. 7.8. A subset of all the detected key-points is selected by using a mask resulting from region segmentation. Each selected key-point is associated with a bag of key-points as in [Csurka et al., 2004]. The bags of key-points are a priori defined for each geometric model during the visual concept learning phase. In this experiment, with have used 15 bags of key-points. This number

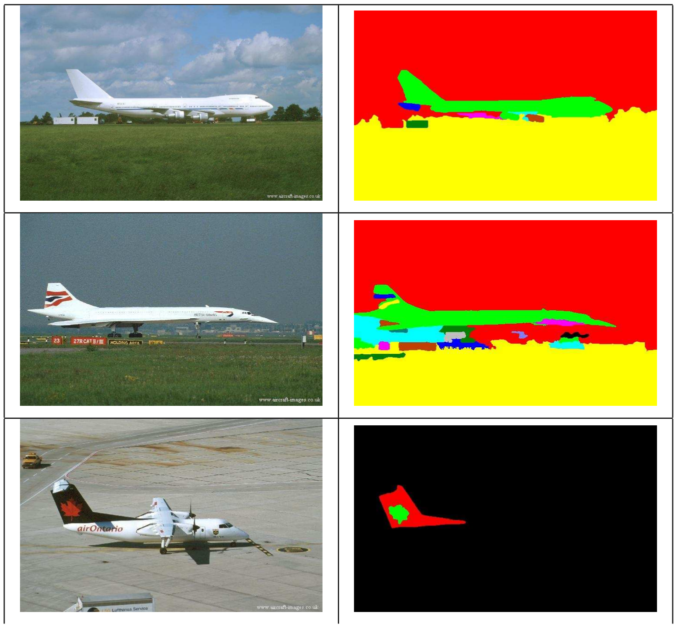


Table 7.10: Examples of automatic image segmentation results.

has been chosen as a good trade-off between *processing speed* and visual concept detection accuracy. This result in a feature vector $\mathbf{x} \in \mathbb{R}^{15}$ that can be used by the following visual concept detectors: $d_{AircraftGeometry}$, $d_{CarGeometry}$, $d_{MotorbikeGeometry}$.

7.4.6 Retrieval

Once, the image database has been indexed, retrieval is initiated by symbolic queries (see fig. 7.9). A Recall/Precision curve has been obtained (by a variation of $comp_{th}$ from 0 to 1 with a variation step of 0.01) for the following domain classes : $Aircraft$, $MotorBike$, Car and Sky (fig. 7.10).

Precision is defined as the ratio between the number of relevant retrieved images and the number of retrieved images. *Recall* is defined as the ratio between the number of relevant retrieved images and the number of relevant images in the image database.

The results obtained show that our methodology leads to efficient indexing : For a recall of 0.5, precision is between 0.75 and 0.78 for the domain classes $Aircraft$, $MotorBike$ and Car and of 0.90 for class Sky. These results show that even with very little effort of knowledge acquisition (12 visual concepts and 6 domain classes), the approach offers both good results and semantic richness.

7.4.7 Implementation Remarks

The software components used for object categorization, indexing and retrieval are implemented in C++. We have used the LAMA platform [Moisan, 1998] for all the developments related to the high-level knowledge and to the visual concept ontology. LAMA is devel-

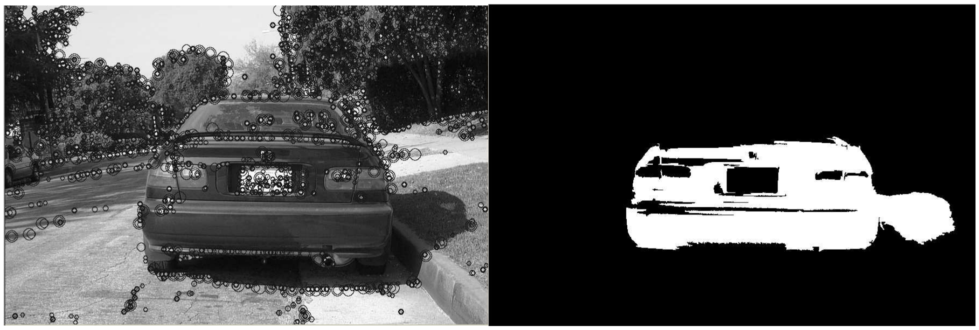


Figure 7.8: Example of Sift features computed on an image containing a car. On the right, the associated segmented image (after region selection). The resulting mask is used to filter the key-points detected in the background: any key-point which is not in the area resulting from segmentation is not taken into account by visual detection.

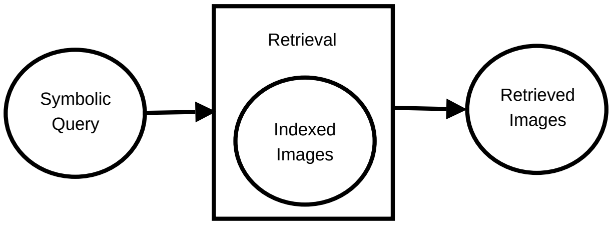


Figure 7.9: Retrieval is achieved by symbolic querying.

oped in the Orion team and is a software platform devoted to the generation of knowledge based systems. It contains reusable software components involved in the use of knowledge based systems (e.g. inference engines, user interfaces, verification tools). In particular, we have used the BLOCKS [Moisan et al., 2001] toolkit which provides generic C++ data structure such as the notions of taxonomy, knowledge base, frame, slot. A new object categorization algorithm has been developed by using these software components.

For the development of the low-level feature extraction and image segmentation algorithms, we have used the LTI-Lib [3], a C++ library developed at the Aachen University of Technology. This library has many different types of algorithms for *image segmentation*, *feature extraction* and *machine learning*. It is well documented, well maintained and has an excellent object-oriented design. We had to extend this library with new low-level feature extraction algorithms which were not available: co-occurence matrices and color coherence vectors. Moreover, for using the Sift features as described in [Csurka et al., 2004], additional development has been required.

[3] http://ltilib.sourceforge.net/

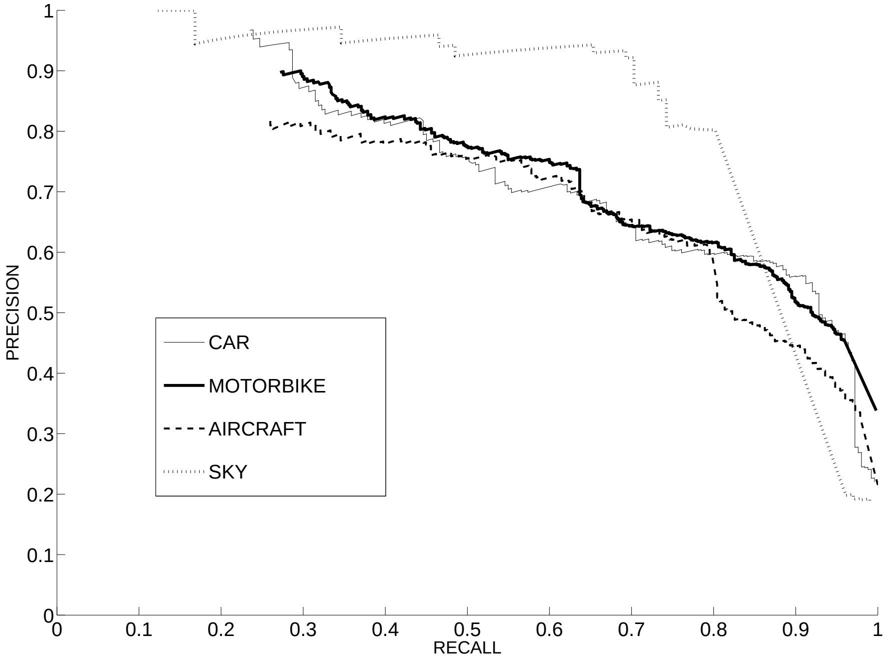


Figure 7.10: Recall/Precision curves obtained for the following domain classes: Car, Motorbike, Aircraft, and Sky.

7.4.8 Conclusion

We have shown how our the proposed phases lead to an efficient semantic image indexing and retrieval system. Our approach does not imply storing neither high-dimensional feature vectors nor segmented image regions. One strength of our approach is the fact that it is very efficient for a given application domain. The semantic richness is also high (e.g. composition and specialization relations) even if the knowledge acquisition effort is low.

On the other hand, the end-user is only able to retrieve the objects of the domain of interest considered. This limit of most a-priori knowledge-based systems is related to the close-world assumption. It is not yet clear how the proposed approach can be generalized to a large number of classes and thus would address the needs of users from different domains of interest.

The visual concept learning process and the image indexing process use several types of features. A combination of classic region segmentation with local (i.e. Sift) features is useful for filtering key-points detected in the background and not detected on the object of interest. Even if segmentation is not perfect, the selected key-points are efficient for characterizing the objects of interest.

7.5 Conclusion

We have applied the proposed approach for achieving three types of tasks: *knowledge acquisition*, *visual concept learning and detection*, and *semantic image indexing and retrieval.* This shows that each phase of our approach has an interest independent from the others.

The knowledge acquisition phase is useful for acquiring and saving expert knowledge in a structured manner. This is particularly important in domains of expertise where the number of experts is declining.

We have shown that the way texture patches are perceived by human could be reproduced with accuracy by visual concept learning and then by visual concept detection.

Our approach has been successfully applied to the problem of semantic image indexing and retrieval. We have shown that the retrieval problem is simplified by the proposed symbolic image indexing approach. The proposed approach does not require too much effort for being both efficient and user-friendly. Moreover, the three levels of abstraction of the image interpretation problem are clearly separated. From a software engineering point of view, this clear separation makes easier the *re-usability* and the *extendability* of the different software components of the implemented system.

Chapter 8

Conclusion

In this thesis, we have investigated the topic of *object categorization.* Our approach relies on *a priori* knowledge formalization and acquisition techniques. The *semantic gap* between high-level knowledge and low-level image features extracted by image processing algorithms is filled by an intermediate level of semantics (*the visual concepts*) combined with machine learning techniques. The proposed approach is composed of three phases:

1. A *knowledge acquisition phase* which consists of acquiring a hierarchy of domain classes described by visual concepts. This knowledge acquisition phase is driven by a visual concept ontology.

2. *A learning phase* which consists of producing a set of visual concept detectors to enable visual concept detection in any image.

3. *An object recognition phase* that uses acquired knowledge and the visual concept detectors to achieve complex object recognition. The proposed object categorization algorithm takes into account object subparts and spatial relations.

The proposed approach has been illustrated in three different applications: knowledge acquisition in the domain of *palynology*, learning and detection of texture concepts, and semantic image indexing and retrieval in the application domain of transport vehicles.

The combination of machine learning techniques with ontological engineering brings convenience and re-usability to the proposed approach.

From the point of view of the framework of *cognitive vision*, our approach is related to the following facets of cognitive vision: *learning*, *recognition*, *representation*, and *reasoning.* Moreover, the proposed approach enables the following functionalities to be achieved: *classification and categorization*, *detection and localization*, *concept formation and visualization.*

8.1 Overview of the Contributions

- **A visual concept ontology** composed of *color*, *texture* and *spatial* concepts, and of *spatial relations*. This ontology enables the description of a wide number of object categories. It is richer than existing comparable ontologies ([Mezaris et al., 2004], [Coenen and Visser, 1999]).

- **A knowledge acquisition methodology**. The knowledge acquisition bottleneck in computer vision has been clearly identified as a problem in [Draper et al., 1996] but has rarely been addressed as a problem as such in the literature. We provide a well-defined methodology for acquiring the knowledge of a domain of interest as a taxonomy/partonomy of domain classes described by visual concepts. Our method *reduces the knowledge acquisition bottleneck* by a combined use of ontological engineering and of machine learning techniques. For instance, no *inference rules* which are difficult to manage have to be defined for enabling the detection of visual concepts in the image.

- **A visual concept learning method** that produces a set of visual concept detectors useful for achieving object categorization. The originality of this approach is the transformation of the object learning problem into a visual concept learning problem.

- **A categorization algorithm**. We have proposed an original object categorization algorithm which performs visual concept detection task and which is able to reject previously unseen visual concepts. Compared to existing object categorization approaches the result of object categorization is in terms of the classes of the domain and also in terms of visual concepts. The categorization results have better semantic richness than appearance based techniques.

- **Application to the problem of semantic image indexing and retrieval**. We have shown how the proposed approach can be applied to the problem of image indexing and retrieval. The symbolic nature of the categorization process enables efficient indexing and retrieval and also bring user-friendliness to the retrieval phase. Indeed, a key-word based approach is natural for many users of text search engines.

8.2 Discussion

Some functionalities are lacking to obtain a complete cognitive vision system as defined in section 2.1: *tracking*, *prediction*, *inter-agent communication and expression*, *visuo-motor coordination*, *embodied exploration*. These functionalities could be achieved by coupling the proposed architecture with a video monitoring platform or by integration into robotic mobile agent.

The current trend in object recognition is the use of local features ([Csurka et al., 2004], [Jurie and Schmid, 2004], [Fergus et al., 2003]). These methods are very efficient for achieving the recognition of manufactured objects with very distinctive parts (e.g. motorbikes, cars). They are not suited to the recognition of uniform objects with smooth edges (e.g. sky). Other techniques such as color coherence vectors are better for that purpose. Our approach should not be compared to the techniques enumerated above but rather be considered as a shell for these techniques. Our approach enables to take the best of many different types of low-level features and algorithms by using them in a focused and structured manner. The structure of use of these low-level features is brought explicitly by the knowledge of the domain of interest. Advances in the domain of feature extraction can be integrated in our framework to improve the overall performance of the categorization system.

The description in terms of visual concepts during knowledge acquisition describes the *appearance* of the objects of interest. For a different context (e.g. change of point of view), visual description has to be at least partially changed. This means that more knowledge has to be provided by the expert. It is not clear how much knowledge can be provided in a reliable way. This aspect makes the approach particularly well-suited to application where the context of acquisition (e.g. point of view, magnification) is well-controlled. For applications where the acquisition conditions are not too variable, it is possible to provide more annotated image samples to take into account the different acquisition conditions.

Another limit of the proposed approach is that visual concepts which are discriminative for an object category cannot always be named explicitly. The proposed approach should be combined with approaches based on a *visual thesaurus* [Fauqueur and Boujemaa, 2003]. This would provide to the user two different ways for accessing the visual concept level. Which way is the more user-friendly remains an open question.

The quality of the results of object categorization depends heavily on the efficiency of image segmentation. It is now accepted that perfect initial object segmentation is impossible. The problem of segmentation *should not* be separated from the problem of interpretation. These two problems are strongly linked and interleaved. The approach used by Leibe who proposes a framework for interleaved object segmentation and interpretation is very promising [Leibe, 2004]. It is clear that the combination of *top-down* and *bottom-up* mechanisms is a necessary condition for solving the problem of object recognition.

From the point of view of the application of our approach to the problem of image indexing and retrieval, what we proposed is really well suited to domain specific image indexing and retrieval problems. The use of a priori knowledge combined with visual concept learning produces good results without providing too much knowledge. Moreover, the approach enables user-friendly querying by keywords. Reaching the same level of semantic richness for the indexing and the retrieval of images from any domain of interest would require a knowledge base with at least *thousands* of domain classes described by visual concepts. The ontological commitment [Bachimont, 2000] required for achieving this goal would be very hard to obtain. This kind of approach has been described in [Hauptmann, 2004]. In this work, the author raises the need of a large enough intermediate semantic layer that would enable the indexing and the retrieval of general-purpose video content. The visual concept ontology proposed in this thesis can be one of the tools used for the creation of this intermediate semantic layer.

8.3 Future Work

8.3.1 Short-term Perspectives

Integration of Visual Data Management Functionalities

The visual data management module specification presented in [Hudelot, 2005] has the interesting functionality of grouping. Another interesting point of view on the problem of grouping can be found in [Zlatoff et al., 2004]. The authors of this work argue that the grouping process has to be aware of what it treats without being completely dependent. Grouping is defined as the process that organizes image data entities into higher level structures. This process is natural for human beings and was the subject of an intensive research in the Gestalt school of psychology. The integration of this functionality would be useful to group the regions resulting from image segmentation into meaningful entities.

Management of Different Types of Image Primitives

One additional element that should be added to the current approach is the management of *different types of image primitives*. For the moment, only *regions* are taken into account by the visual concept learning phase and by the object categorization phase. *Edges* should also be taken into account so as to enable the learning and the recognition of lineic objects (e.g. roads in aerial imaging).

Evaluation of the Usefulness of the Proposed Approach for a Computer Vision Expert

We have shown that our approach brings important improvements at the level of user-friendliness. The visual concept ontology enables non specialist in computer vision to access complex low-level image processing algorithms and features. The framework should be provided to a computer vision expert (i.e. the knowledge acquisition tool, the learning categorization engine) to see how the approach is useful for such an expert. The following questions should be asked:

- Does the approach brings enough flexibility for answering the computer vision expert needs?
- How can the resulting categorization system can be interfaced with other software architectures (e.g. a video surveillance platform)?
- How much is the efficiency of the computer vision expert improved compared to the use of his/her favorite computer vision library?

Graphical Representation of the Visual Concepts

At the implementation level, the knowledge acquisition tool Ontovis should be improved in order to enable visual concept visualization during knowledge acquisition. For the moment, only the labels of the visual concepts are displayed in a *textual* form. A graphical display (e.g. icons) of the visual concepts would improve the usability of this tool. This perspective is related to *semiotics* [1] issues. The graphical representation of some visual concepts is not obvious. For instance, the concept $GranulatedTexture$, has many possible graphical representations. Currently, visual concepts can be combined as *disjunction* or *conjunction.* Further research is required to see how the visual concepts can be combined graphically. This approach can be brought close to the *query by sketch paradigm* popular in the image retrieval community. An example of such approach is the work presented in [Chang et al., 1998] where the notion of *semantic visual templates* is used for querying purposes.

Improvement of the unknown object/class matching approach

We have seen in chapter 6 that the computation of the compatibility of an unknown object with a domain class is based on a weighted sum taking into account the visual description of the class and the visual concepts detected in the image. With the current approach, for two classes with the same visual description, if the result of the weighted sum is greater that the predefined threshold $comp_{th}$, the classes cannot be distinguished. The notion of importance associated with the attributes of the domain classes has been introduced for coping with this problem. Another possibility is to introduce a new visual concept in the visual concept ontology so as to obtain two different visual descriptions. The importance of each attribute is not always easy to guess. Optimization techniques could be used to obtain the importance of the attributes in order to maximize the separability of the classes. This process would use a set of images annotated in terms of visual concepts and in terms of domain classes.

Use of Semantically Labeled 3-D Meshes

We have seen in section 5.2.2 that 3-D meshes can be used to obtain 2-D projections of the objects of interest. The 2-D projections are then labeled by visual concepts. Currently 3-D meshes are used in their whole for obtaining the 2-D projections. 3-D designers often associate labels to 3-D primitives used for the modeling of the objects they design. (e.g. the label *wheel* can be associated with the *cylinder* used for modeling the wheel of car). It would be interesting to use these labels for obtaining 2-D projections of specific subparts of the objects of interest. The annotation of the 2-D projections of object subparts would then be made possible.

[1] Semiotics is defined as the study of signs, both individually and grouped in sign systems, and includes the study of how meaning is transmitted and understood.

Use of Low-Level Features for Learning and Detecting Complex Spatial Relations

The formalism introduced in chapter 4 enables binary relations between objects to be described. This should be improved by introducing the *n-ary* relations between three or more objects. This implies that low-level features will be required for the detection of these relations. As explained in [Scott et al., 2005], the use of Matsakis histograms [Matsakis et al., 2004] is efficient for the real-time low-level extraction of spatial relations involving several objects in a manner insensitive to *scaling*, *rotation*, and *translation* of the spatial configuration.

Improvement of the Manual Segmentation Phase

In chapter 5, the interest of semi-automatic segmentation methods has been stressed. In particular, the technique called the *intelligent scissors* has proved to be useful for achieving accurate object segmentation. A new method called *Lazy Snapping* [Li et al., 2004] seems to be even more efficient. Lazy Snapping separates coarse and fine scale processing, making object specification and detailed adjustment easy. Moreover, Lazy Snapping provides instant visual feedback, snapping the cutout contour to the true object boundary efficiently despite the presence of ambiguous or low contrast edges. Instant feedback is made possible by a novel image segmentation algorithm which combines graph cut with pre-computed over-segmentation. Usability studies indicate that Lazy Snapping provides a better user experience and produces better segmentation results than the state-of-the-art interactive image cutout tools. Another technique called Grabcut [Rother et al., 2004] also leads to very efficient object segmentation while requiring very little user interaction: the user only has to drag a rectangle around the object of interest and then the object is accurately automatically segmented.

Towards Adaptive Image Segmentation

Chapter 7 has shown the importance of segmentation results. Much progress has to be done at this level. A work currently conducted in the Orion team by V. Martin [Martin et al., 2006] deals with the problem of *adaptive image segmentation*. In this work, a scheme is proposed to automatically select segmentation algorithm and tune their key parameters thanks to a preliminary supervised learning stage. The approach is enabled by a set of manually segmented images (i.e. a ground-truth). This learning stage is based on machine learning techniques and is composed of two steps: *optimal parameters computation by optimization* and *algorithm selection learning*. At the end of the learning stage, all the manually segmented images are associated with the set of optimal parameters and a segmentation algorithm form a case base. The automatic segmentation phase uses the results of the learning stage for achieving adaptive segmentation (see figure 8.1). Low-level features extracted from the image are given in input of the algorithm predictor trained in the learning stage (1). Then, similarity is determined by looking up the case base for similar cases (2). When the closest one is found, the image is segmented with corresponding optimal parameters. This approach has given interesting preliminary results and should be integrated in the framework proposed in this thesis to obtain better segmentation results.

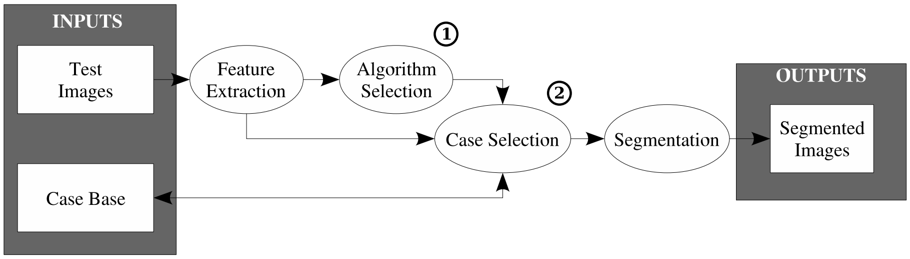


Figure 8.1: Towards adaptive image segmentation. After a learning stage involving manually segmented images, test images are segmented in an adaptive manner (i.e. automatic parametrization and algorithm selection).

8.3.2 Long-term Perpectives

Cooperation with VSIP

The Video Surveillance and Interpretation Platform (VSIP) [Avanzi et al., 2005] developed in the ORION team is used for the analysis of the behavior of many different types of *moving objects* (e.g. cars, people). Currently, the recognition of the different categories of objects is based on simple low-level features such as the *width* or the *height* of the moving objects. This could be improved by interfacing the video surveillance platform with our object recognition platform. An *off-line* phase would consist of using the knowledge acquisition tool, Ontovis, to define the objects involved in the scenarios to recognize and to provide image samples of these objects. During scenario recognition, the video surveillance platform could send *categorization requests* and receive *categorization results* answering categorization needs (fig. 8.2). This approach would give a system capable of the following functional capabilities of cognitive vision systems (see section 2.1.2): *detection and localization*; *tracking*; *classification and categorization*; *concept formation and visualization*.

Combination of query by example with query by concept

Chapter 7 has shown how the proposed approach is used for semantic image indexing and retrieval purposes. In particular, the set of indexed images can be queried in terms of domain classes and in terms of visual concepts. This approach to the image retrieval problem is a *query by concept* paradigm. This approach could be combined with the *query by example* paradigm. The query by example formalism cannot capture the conceptual essence of one image or even several image samples [Town and Sinclair, 2004]. Our approach could

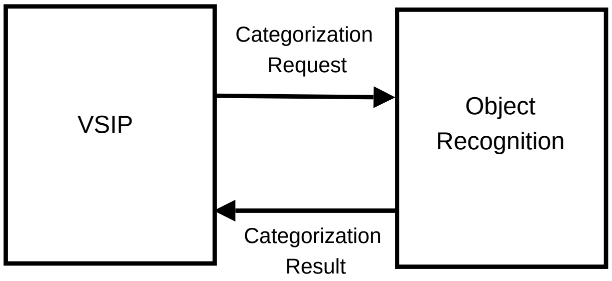


Figure 8.2: The Video Surveillance and Interpretation Platform (VSIP) of the ORION team could use the object categorization functionalities offered by our framework.

be used to derive the symbolic description of the query image by object categorization so as to produce a symbolic query. This would consist of transforming a query image (or a sub-image) into a set of visual concepts. Then, retrieval would be based on the symbolic description of the query image (fig. 8.3).

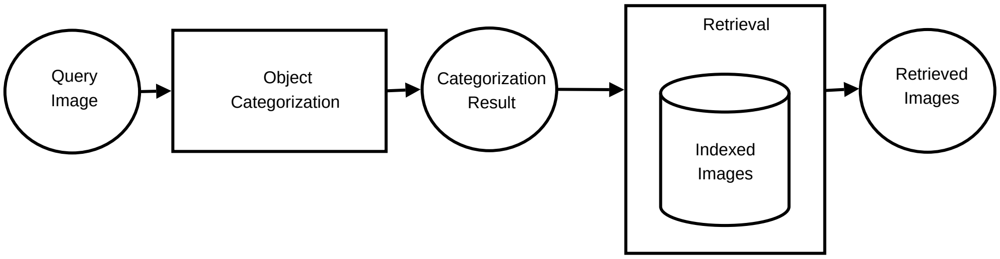


Figure 8.3: Combination of the *query by example* paradigm with the *query by concept* paradigm. The query image is first used as an input of the object categorization process. The categorization result is used for querying the indexed images.

Improvement of the Annotation Process

As explained in chapter 5, the image sample segmentation and annotation process is done at the visual concept level: the result of this process is a set of image regions labeled by visual concepts. Visual concepts and not domain classes are used for annotation. From the user-friendliness point of view, a class-based annotation process would be better than a visual concept based annotation process (i.e. the number of annotations would be lower). The following trade off would be interesting to reach: a part of the samples could be *completely* annotated by visual concepts (e.g. *Blue*, *Gray*) *and* by class name (e.g. *Sky*). The other part of the samples would then be *partially* annotated only by class names. The corresponding visual concepts could then be obtained to form a complete annotation by *analogy* with the complete annotations performed on a subpart of the image database.

Relevance Feedback

One important improvement that could be brought to the proposed approach is the introduction of *relevance feedback* mechanisms. We have shown that the categorization result associated with an image is a set of detected visual concepts compatible with a class of the domain taxonomy. The following errors occurs in the categorization result: false positive and false negative at the class level and also at the visual concept level. An interactive step could be integrated to the categorization phase. This step would consists of interactively modifying the unknown object contained in the categorization result. The relevance feedback can be done at two levels: at the visual concept level and at the class level. This interactive step should enable the user to:

- Add to the resulting unknown object the visual concepts not detected in the image (i.e. false negatives).
- Remove some visual concepts (i.e. false positives).
- Modify the validated hypothesis associated with the unknown object (i.e. changing the name of the recognized class).

Dynamic Knowledge Bases

The object categorization process is driven by acquired knowledge. An object in an image can be recognized only as an instance of a domain class of the domain taxonomy. This a limit of the *close-world* assumption made by most knowledge-based systems. Chapter 6 has introduced the notion of *reject* in case of the occurrence of previously unseen visual concepts. In some cases, several visual concepts might be rejected at the same time. It would be interesting to study the statistical co-occurrence of the reject of several visual concepts. Some specific configurations of rejected visual concepts may correspond to a new domain class. The domain class resulting from recurrent rejects of the same set of visual concepts may then proposed to the expert for labeling purposes by showing her/him the regions which have entailed rejection of the visual concepts.

Incremental Learning

Currently, when a new sample is added to the set of annotated regions, the learning process described in chapter 5 has to be performed completely. The underlying optimization process of the training of a Support Vector Machine takes into account all the feature vectors of the training set. The use of incremental machine learning techniques would be useful for obtaining the property of continuous learning. An *incremental Bayesian model* such as the one used in [Fei-Fei et al., 2004] could be used. The performance of this incremental model are comparable to a *batch Bayesian model*.

Video Analysis and Retrieval

The keynote talk of A. Hauptmann [Hauptmann, 2005] at CIVR 2005 has shown that there is still a lot to do in the domain of video analysis and retrieval. Concerning the possible extensions of this thesis, an important research direction which has to be explored is the extension of the visual concept ontology with temporal concepts. These kinds of concepts would be useful to describe the movement of the objects of interest. For instance, the trajectory of mobile objects such as cars can be described by geometric concepts. Temporal concepts for describing the morphological evolution of the objects (e.g. cell evolution in cytology) should also be added in the ontology. To apply the approach to generic videos (e.g. news), the problem of motion estimation and trajectory representation in uncalibrated videos [Nunziati et al., 2005] will have to be tackled.

Generic Image Database Indexing and Retrieval

For image indexing and retrieval applications, the proposed approach is particularly suited to domain specific applications (e.g. indexing and retrieval of images in the domain of transport vehicles). Many concepts of the visual concept ontology are valid for different types of applications (e.g. concepts of type *Hue*). The indexation of generic image databases (i.e. containing images of many different domains of application) could be enabled with the training of the visual concept detectors corresponding with these generic visual concepts with images from different types of applications. The image database would then be indexed only in terms of visual concepts. Therefore, querying would only be possible in terms of visual concepts. This means that the conceptual level of querying would be at an intermediate level of semantics. It would still be useful for retrieving categories of images which have a particular characteristic visual appearance.

Better Use of Visual Concepts for Guiding Segmentation

In chapter 6, we have seen that the *Position* and the *Size* visual concepts are involved in the automatic image segmentation and region selection process. Some work has to be undertaken in order to take other categories of visual concepts into account (e.g. texture visual concepts). This can be achieved in program supervision framework [Thonnat et al., 1999] or by a rule-based approach.

Use of Geons for Better Object Modeling

The notion of Geon was introduced by Biederman in [Bierderman, 1987] and then brought to the computer vision community. Geons are simple geometric primitives that can be used for modeling many different types of objects. The principal of use of Geons is illustrated in fig. 8.4. The potential of geons for generic 3-D Object Recognition is presented in [Dickinson et al., 1997]. Geons have been used for true 3D object recognition [Borges and Fisher, 1996]. A method for approximating 3-D shapes using parametric geons is presented in [Wu and Levine, 1997]. The geons could be integrated in the visual concept ontology so as to enable the experts to model easily complex 3-D objects. The problem of reliable geon extraction from 2-D images is still an open problem.

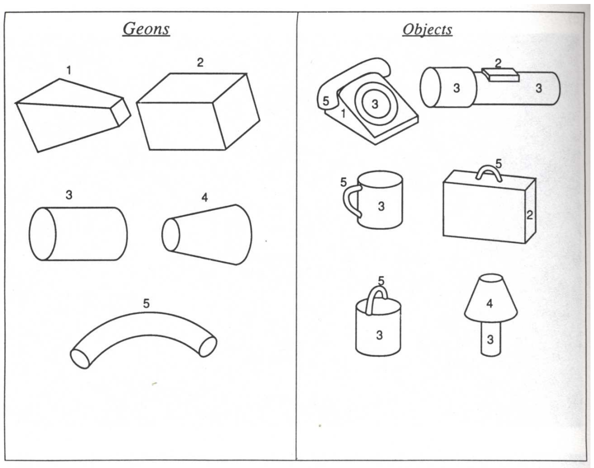


Figure 8.4: Biederman's Geons [Bierderman, 1987] and some objects modeled with these Geons.

Application to Range Images

Range images can be obtained by stereo sensors. The result of data acquisition is a depth map. 3-D Visual concepts dedicated to the description and to the recognition of the appearance of objects visualized in range images would be useful (e.g. in robotic applications). An overview of object recognition in dense-range images can be found in [Arman and Aggarwal, 1993]. A lot of work would have to be done at the level of segmentation and feature extraction. 3-D spatial relations would also have to be included in the ontology in order to describe the spatial relations between objects in a 3-D world.

Appendix 1 : Publications of the Author

- **International Journals with Peer-review:**

  [1] Ontology Based Complex Object Recognition. Maillot, N., Thonnat, M. Image and Vision Computing Journal. Under Minor Revision.

  [2] Towards Ontology Based Cognitive Vision (Long Version). Maillot, N., Thonnat, M., Boucher, A. Machine Vision and Applications Journal. Springer-Verlag Heidelberg. Dec. 2004, 16(1), pp 33–40.

- **International Conferences with Peer-review:**

  [1] A Learning Approach for Adaptive Segmentation. Martin, V., Maillot, N., Thonnat, M. In Proceedings of the Fourth International Conference On Computer Vision Systems (ICVS 2006), New York, USA, 5-7 Jan. 2006.

  [2] A Weakly Supervised Approach for Semantic Image Indexing and Retrieval. Maillot, N., Thonnat, M. In Proceedings of the International Conference on Image and Video Retrieval (CIVR 2005), Singapore, 20-22 Jul. 2005.

  [3] Ontology Based Object Learning and Recognition : Application to Image Retrieval. Maillot, N., Thonnat, M., Hudelot, C. In Proceedings of the 16th IEEE International Conference on Tools for Artificial Intelligence (ICTAI 2004). Boca Raton, Florida, 15-17 Nov. 2004.

  [4] Towards Ontology Based Cognitive Vision. Maillot, N., Thonnat, M., Boucher, A. In Proceedings of the Third International Conference On Computer Vision Systems (ICVS 2003), Graz, Austria, Apr. 2003.

- **Technical Reports:**

  [1] Ontologies For Video Events. Brémond, F., Maillot, N., Thonnat, M., Van-Vu, T. INRIA Research Report (RR-5189)

Appendix 2 : French Introduction

La reconnaissance automatique d'objets est un sujet de recherche étudié depuis trois décennies. Malgré certains succès dans des domaines comme le contrôle qualité ou la reconnaissance de visages, le problème de la reconnaissance automatique d'objets reste ouvert. La plupart des systèmes actuels souffrent d'un manque de flexibilité et d'adaptabilité. Ce constat est vrai pour nombre de systèmes de vision par ordinateur.

Dans l'objectif d'améliorer la situation, une discipline appelée *vision cognitive* a été introduite. Un plan de travail relatif à cette discipline peut être trouvé dans [Auer et al., 2005]. Ce document explique que le terme *vision cognitive se rapporte à la volonté de mettre au point des systèmes de vision plus robustes et plus adaptables ayant les facultés cognitives suivantes : l'apprentissage, l'adaptation, l'évaluation d'alternatives ou même la capacité à développer de nouvelles stratégies d'interprétation et d'analyse.*

Cette thèse est focalisée sur le problème de la reconnaissance d'objets. Nous abordons ce problème en tenant compte des fondations suivantes de la vision cognitive : reconnaissance, architecture, représentation, apprentissage et communication. Ce manuscrit montre comment ces fondations peuvent être utilisées pour construire un système opérationel de reconnaissance d'objets.

Des avancées dans le domaine de la reconnaissance d'objets ont des conséquences dans les domaines suivants:

- **Indexation et recherche d'images et de vidéos**. L'importante quantité de contenus multimédias produits par les appareils photographiques personels, par les enregistreurs vidéos ou encore disponibles sur le World Wide Web créée un besoin urgent pour des techniques efficaces d'accès à ces contenus.

- **Vidéo Surveillance**. La vidéo surveillance automatique passe par la reconnaissance de scénarios impliquant des objets complexes (e.g. personne, voiture). La plupart des algorithmes de reconnaissance de scénarios (e.g. [Vu et al., 2003]) font l'hypothèse que les objets impliqués dans la scènes sont correctement reconnus. Cela signifie que des algorithmes fiables de reconnaissance d'objets sont requis pour que ces algorithmes produisent des résultats satisfaisants.

- **Robotique**. Les capacités de reconnaissance de scènes et d'objets sont très importantes pour les systèmes robotiques. Dans [Auer et al., 2005], il est expliqué qu'un système robotique mobile doit être capable d'acquérir un modèle de l'environnement physique, d'identifier des objets et de comprendre leurs fonctions mais aussi de détecter de nouveaux objets. Des fonctionalités avancées d'identification et de reconnaissance d'objets sont nécessaires pour réaliser ces tâches.

- **Environnements Intelligents**. Dans [Campbell and Krumm, 2000], l'importance de la reconnaissance d'objets dans le contexte des pièces intelligentes est présentée. A la maison ou au bureau, des persones inter-agissent entre elles mais aussi avec de nombreux objets (e.g. clavier, télécommande, téléphone, livre). Un exemple de fonctionalité d'une pièce intelligente est la capacité à retrouver des objets perdus (e.g. clés). Une autre fonctionalité importante est la capacité à inférer les intentions de l'utilisateur (e.g. allumer la lumière lorsqu'une personne commence à lire un livre). Dans les deux cas, des mécanismes de reconnaissance d'objets sont requis.

8.4 Contexte de l'étude

Ce travail a été mené dans l'équipe Orion de l'Inria Sophia Antipolis. L'équipe Orion est une équipe multi-disciplinaire à la frontière de la vision par ordinateur, de l'intelligence artificielle, et du génie logiciel. L'équipe a accumulé une forte experience dans ces domaines au cours des années.

L'un des centres d'intérêt de l'équipe est l'interprétation d'images de vidéos à base de connaissances. Dans [Thonnat and Bijaoui, 1989], une approche à base de connaissances est utilisée pour la reconnaissance automatique de galaxies. Dans [Vu et al., 2003], un nouvel algorithme temps-réel de reconnaissance de scénarios est présenté.

Des contributions ont aussi été faites par l'équipe dans le domaine du génie logiciel. En particulier, le *pilotage de programmes*, a été introduit pour améliorer la gestion de librairies de traitement d'images [Shekhar et al., 1998]. Dans [Ossola et al., 1996], Ossola propose une plate-forme logicielle basée sur la coopération de deux systèmes à base de connaissances. Le premier est dédié à l'interprétation d'images haut niveau. Le second est dédié au pilotage de programmes. Plus récemment, dans [Hudelot, 2005], une plate-forme de vision cognitive est présentée. Cette dernière contient un système à base de connaissances dédié à la gestion de données images. Ce système opère en coopération avec deux autres systèmes respectivement dédiés à l'interprétation d'images haut niveau et au pilotage de programmes.

8.5 Problème et Objectifs

Ce travail se place dans le cadre de l'interprétation sémantique d'images et plus particulièrement dans celui de le reconnnaissance d'objets. La difficulté de l'interprétation sémantique d'images est illustrée par la figure 8.5. En effet, cette image peut être interprétée comme un objet clair sur un fond sombre. Il est également possible de percevoir cette image comme un objet astronomique ou plus précisément comme une galaxie spirale. Cette exemple illustre le fait que l'interprétation sémantique d'image repose sur une connaissance a priori. Ceci signifie que la sémantique n'est pas dans l'image mais résulte de l'association de la perception avec la connaissance a priori acquise au cours de l'expérience. D'une personne à l'autre, en fonction de son expérience, cette associatiation peut être faite différement. Ceci engendre le phénomène de la polysémie de l'image. Comme expliqué dans [Shatford, 1986], "la joie et la frustration des resources images proviennent du fait qu'une image peut signifier plusieurs choses pour différentes personnes". Dans [Barthes, 1977], Barthes différencie le sens dénoté du sens connoté. Le sens dénoté d'une image correspond à la réalité perçue. Le sens connoté correspond à la somme des significations possibles de l'image. Ces différentes significations possibles sont dues à des facteurs historiques ou culturels. Une étude intéressante de l'influence de la polysémie de l'image dans le domaine de la recherche de vidéos est proposée dans [Christel and Hauptmann, 2005].

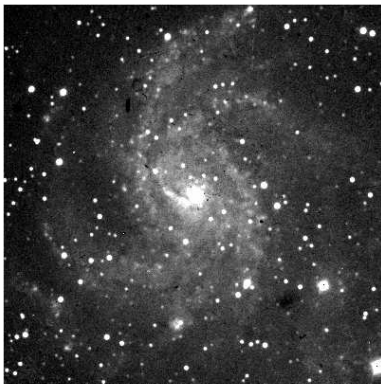


Figure 8.5: L'interprétation sémantique de cette image demande des connaissances en astronomie.

La figure 8.6 monte l'importance toute particulière du contexte. Cette figure a été utilisée lors d'une campagne publicitaire du journal *Le Monde* qui insiste sur le fait que le sens d'une image est en dehors de cette image (i.e. dans ce cas, le sens provient du texte écrit par les journalistes). Cette figure montre trois images du même événement. L'image du haut peut être interprétée comme une poignée de main qui est un code et une règle de politesse. Il est difficile d'en dire plus. Dans l'image du milieu, il est possible d'en dire plus. Il semble que la scène corresponde à un accord entre trois personnes. L'une d'entre elles porte un foulard. On peut se demander s'il s'agit un accord commercial dans le domaine du pétrole. L'image du bas a un sens completement différent. La présence de Bill Clinton combinée avec cette poignée de mains donne un sens différent à cette image : en tenant compte du contexte politique, cette image peut être interprétée comme le résultat d'une négociation relative à des problèmes politiques complexes. Cette interprétation demande des connaissances en politique internationale. Cette image pourrait être montrée à un enfant qui n'en dirait probablement pas plus qu'un adulte cultivé sur l'image du milieu.

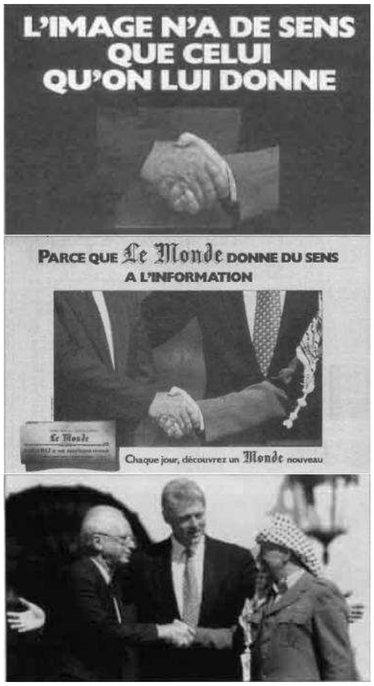

L'IMAGE N'A DE SENS
QUE CELUI
QU'ON LUI DONNE
PARCE QUE Le Monde DONNE DU SENS
A L'INFORMATION
Le Monde
Chaque jour, découvrez un Monde nouveau

Figure 8.6: Trois images du même événement: les accords d'Oslo (1993). L'identification des personnes dans l'image du bas lui donne une signification politique particulière.

Ces observations amènent les questions suivantes:

- Comment acquérir la connaissance a priori pour permettre des tâches de reconnaissance utiles (i.e. répondre aux besoins de l'utilisateur dans un contexte donné) ?
- Comment reproduire le processus expérimental qui ancre la connaissance avec la perception ? Une importante question sous-jacente est de réduire le fossé sémantique entre la connaissance haut niveau et les descripteurs image bas niveau extraits par des algorithmes de traitement d'images.
- Comment utiliser cette connaissance ancrée pour permettre une reconnaissance efficace ? En d'autres termes, une fois que cette connaissance a été acquise et ancrée, comment l'utiliser comme support de la reconnaissance d'objets ?

Notre objectif est de résoudre les problèmes ci-dessus en apportant les contributions suivantes :

- Nous proposons d'utiliser les avancées faites par la communauté de l'ingénierie des connaissances pour réduire le goulot d'étranglement constitué par l'acquisition des connaissances. Une ontologie de concepts visuels est proposée afin de réduire le fossé sémantique. Cette ontologie contient des concepts visuels utiles pour la description des objets d'intérêt.

- Une approche pour ancrer la connaisance haut niveau avec les données images est aussi proposée. Cette approche repose sur des techniques d'apprentissage supervisé et non-supervisé. Cette approche consiste à apprendre la description visuelle des classes d'intérêt. Ceci signifie que les classes d'intérêt sont apprises par le biais d'une couche sémantique intermédiaire.

- Un algorithme de reconnaissance d'objet qui utilise la connaissance apprise est également introduit. Cet algorithme génère des hypothèses à vérifier dans l'image à interpréter.

Un point important de notre travail est que nous plaçons ces contributions dans le cadre de la vision cognitive. Plus précisement, nous montrons comment ce cadre permet une combinaison efficace de paradigmes de vision à base de connaissances et de vision par apparence.

8.6 Structure du Manuscrit

Ce manuscrit est structuré en six chapitres.

Le chapitre 2 présente la vision cognitive, ses fondations scientifiques et les défis qu'elle amène. Cette discipline émergente est utilisée comme cadre de notre travail qui traite du problème de la reconnaissance d'objets. Ce chapitre présente également les approches existantes au problème de la reconnaissance d'objets: les approches géometriques, par apparence et à base de connaissances. Chacune de ces approches possède des points forts et des faiblesses. Notre objectif et de les utiliser dans un cadre coopératif afin de faire des avancées dans le domaine de la reconnaissance d'objets. Un élément clé de cette coopération est l'ingénierie ontologique. C'est pourquoi une vue d'ensemble de ce domaine est également proposée.

Le chapitre 3 présente nos objectifs ainsi qu'un aperçu de notre approche. Le processus d'interprétation d'images repose sur une connaissance a priori combinée à de l'information sensorielle. Notre approche consiste d'abord à acquérir la connaissance a priori. Ce processus d'acquisition de connaissances est facilité par l'ingénierie ontologique. Le résultat de cette phase d'acquisition de connaissances est une taxonomie/partonomie de classes du domaine accompagnées de leur description visuelle. Un autre élément important dans notre approche est l'utilisation de techniques d'apprentissage artificiel qui permettent d'apprendre de la connaissance acquise avec des images d'exemples annotées.

Le chapitre 4 présente en détail la phase d'acquisition de connaissances. Cette acquisition de connaissance est faite par interaction avec l'expert du domaine. Les experts d'un domaine sont rarement compétents en vision par ordinateur. Par contre, ils sont souvent capables de fournir une description visuelle précise des objets de leur domaine. La phase d'acquisition des connaissances consiste à réaliser les tâches suivantes :

- Acquisition de la taxonomie du domaine : cette connaissance du domaine contient les relations de spécialisation et les relations de sous-parties entre les classes du domaine. Cette connaissance est partagée par les experts du domaine et reste facile à acquérir. Cette connaissance est indépendante des couches vision (e.g. contexte d'acquisition).

- Description visuelle des classes d'objets guidée par une ontologie de concepts visuels: cette connaissance est dépendante du contexte d'acquisition.

Le chapitre 5 détaille comment la détection des concepts visuels utilisés pendant la phase d'apprentissage est apprise. Cette phase d'apprentissage consiste à produire un ensemble de détecteurs de concepts visuels. Ceci de trois façons possibles : de manière supevisée et par segmentation manuelle d'images d'exemples, par l'utilisation de modèles 3-D, ou encore de manière semi-supervisée par une combinaison de techniques d'apprentissage supervisé et non-supervisé.

Le chapitre 6 détaille la phase de catégorisation qui utilise le système construit durant les phases d'acquisition de connaissances puis d'apprentissage. Les structures d'entrées/sorties du moteur de catégorisation sont détaillées. La richesse sémantique apportée par la connaissance a priori rend les résultats de catégorisation explicites. En effet, les résultats de catégorisations sont exprimés en termes de concepts visuels mais aussi en termes de catégories.

Le chapitre 7 présente des résultats obtenus en appliquant l'approche proposée. Ce chapitre présente des résultats d'acquisition de connaissances dans le domaine de la palynologie (i.e. l'étude des grains de pollen). Des résultats de catégorisation d'images de texture sont aussi montrés. La troisième partie de ce chapitre montre comment l'approche proposée a été utilisée pour des besoins d'indexation et de recherche sémantique d'images dans le domaine des véhicules de transport.

Le chapitre 8 conclut ce manuscrit en détaillant les contributions mais aussi en proposant des pistes futures de recherche.

Appendix 3 : French Conclusion

Dans cette thèse, nous avons étudié le problème de la catégorisation d'objets. Notre approche repose sur des techniques d'acquisition et de formalisation de connaissances. Le fossé sémantique entre la connaissance haut niveau et les données images est réduit par l'introduction d'une couche sémantique intermédiaire combinée avec des techniques d'apprentissage artificiel. L'approche proposée est composée de trois phases :

1. Une phase d'acquisition de connaissances qui consiste à acquérir une taxonomie/partonomie de classes du domaine accompagnées de description visuelle. Cette phase d'acquisition de connaisances est guidée par une ontologie de concepts visuels.

2. Une phase d'apprentissage qui consiste à produire un ensemble de détecteurs de concepts visuels afin de permettre la détection des concepts visuels utilisés durant la phase d'acquisition de connaissances.

3. Une phase de catégorisation qui utilise la connaissance acquise ainsi que les détecteurs de concepts visuels produits lors de la phase d'apprentissage. L'algorithme de catégorisation prend en compte les sous-parties ainsi que les relations spatiales.

L'approche proposée a été utilisée dans le cadre de trois types d'applications. Tout d'abord pour des besoins d'acquisition de connaissances dans le domaine de la palynologie. Ensuite pour l'apprentissage et la détection d'images de textures. Enfin, pour des besoins d'indexation et de recherche sémantique d'images.

Du point de vue du cadre de la vision cognitive, notre approche est liée aux aspects suivants de cette discipline: apprentissage, reconnaissance, représentation, et raisonnement. De plus, l'approche proposée permet de réaliser les fonctionalités suivantes : classification et catégorisation, détection et localisation, formation de concepts, et visualisation.

8.7 Vue d'ensemble des contributions

- Une ontologie de concepts visuels composée de concepts de couleur, de texture, de forme, ainsi que de relations spatiales. Celle-ci permet la description d'un nombre d'objets d'intérêts. Cette ontologie est plus riche que d'autres ontologies comparables ([Mezaris et al., 2004], [Coenen and Visser, 1999]).

- Une méthodologie d'acquisition de connaissances. L'acquisition de connaissances est un problème difficile [Draper et al., 1996] qui a été rarement traité en tant que tel dans la communauté de la vision par ordinateur. Nous proposons une méthodologie claire pour acquérir la connaissance d'un domaine en tant qu'une taxonomie/partonomie de classes du domaine décrites par des concepts visuels. Notre approche réduit le problème de l'acquisition de connaissances par l'utilisation combinée de l'ingénierie ontologique et de techniques d'apprentissage artificiel.

- Une méthode d'apprentissage de concepts visuels est également proposée. Cette méthode permet de construire un ensemble de détecteurs de concepts visuels qui facilite le problème de la reconnaissance d'objets. L'originalité de cette approche vient du fait que le problème de l'apprentissage d'objets est transformé en un problème réduit à l'apprentissage de concepts visuels.

- Un algorithme de catégorisation est également proposé. Cet algorithme permet la reconnaissance d'objets par l'intermédiaire de la détection de concepts visuels. Les concepts visuels non connus sont rejetés. Comparé à d'autres techniques de catégorisation d'objets, le résultat de catégorisation est exprimé en termes des classes du domaine mais aussi en termes de concepts visuels. Les résultats de catégorisation ont une richesse sémantique plus grande que ceux produits par les techniques de vision par apparence.

- L'application de l'approche proposée au problème de l'indexation et de la recherche d'images permet une indexation efficace et une recherche conviviale. En effet, la recherche se fait à l'aide de mots-clés, ce qui est naturel pour les utilisateurs des moteurs de recherche de documents textuels.

8.8 Discussion

Certaines fonctionalités manquent afin d'obtenir un système de vision cognitive répondant à la définition de la section 2.1: suivi, prédiction, communication inter-agent et expression, coordination visuo-moteur, exploration embarquée. Ces fonctionalités pourraient être obtenues par le biais d'un couplage avec une plate-forme de vidéo surveillance ou par intégration dans un système robotique mobile.

La tendance actuelle dans le domaine de la reconnaissance d'objets est l'utilisation de descripteurs locaux ([Csurka et al., 2004], [Jurie and Schmid, 2004], [Fergus et al., 2003]). Ces méthodes sont très efficaces pour la reconnaissance d'objets manufacturés ayant des sous-parties distinctives (e.g. voitures). Ces techniques ne sont pas adaptées à la reconnaissance d'objets uniformes (e.g ciel). D'autres techniques sont plus adaptées pour cela. Notre approche ne doit pas être comparée directement aux techniques pré-citées mais doit plutôt être considérée comme un moyen de les combiner en tenant compte d'une connaissance a priori. Notre approche combine les avantages des techniques de vision à base de connaissances avec les avantages des techniques de vision par apparence. Des avancées dans le domaine de l'extraction de descripteur peuvent être intégrées au sein de notre approche afin d'améliorer les performances de catégorisation.

La description en termes de concepts visuels durant la phase d'acquisition de connaissances décrit l'apparence des objets d'intérêts. Dans un autre contexte d'acquisition (e.g. changement de point de vue), la description visuelle doit être au moins partiellement changée. Ceci implique que plus de connaissances doivent être données par l'expert. Cette approache est bien adaptée aux applications pour lesquelles les conditions d'acquisition (e.g. grossissement) sont bien controllées. Si les conditions d'acquisition ne sont pas trop variables, alors fournir plus d'images d'exemples peut suffir à prendre en compte les changements de conditions d'acquisition.

Une autre limite de l'approche proposée est que les concepts visuels caractéristiques de certaines classes ne peuvent pas toujours être nommés explicitement. Notre approche pourrait être combinée avec l'approche présentée dans [Fauqueur and Boujemaa, 2003] qui est basée sur un thesaurus visuel. Ceci permettrait à l'utilisateur d'accéder aux concepts visuels de plusieurs façons. La question reste de savoir quelle approche est la plus conviviale.

La qualité des résultats de catégorisation dépend fortement de l'efficacité des algorithmes de segmentation d'images. Il est maintenant connu qu'obtenir une segmentation initiale parfaite est impossible. Le problème de la segmentation ne devrait pas être séparé du problème de l'interprétation. Ces deux problèmes sont liés. Une combinaison de mécanismes *top-down* et *bottom-up* est une condition nécessaire pour résoudre le problème de la reconnaissance d'objets.

Du point de vue de l'application de notre approche au problème de l'indexation et la recherche sémantique d'images, ce que nous proposons est bien adapté à des domaines d'application dédiés. L'utilisation combinée de connaissances a priori produit de bons résultats sans fournir une grande quantité de connaissances. De plus, cette approche permet la recherche par mot-clés. Atteindre un tel niveau de richesse sémantique pour des images de n'importe quel domaine d'application demanderait une base de connaissances constituée d'au moins de milliers de classes. L'engagement ontologique [Bachimont, 2000] nécessaire serait difficile à obtenir. Ce type d'approche a été décrit dans [Hauptmann, 2004]. Dans ce travail, les auteurs soulèvent le besoin d'une couche sémantique intermédiaire qui permettrait l'indexation et la recherche de vidéos de n'importe quel domaine d'application. L'ontologie de concepts visuels proposée dans cette thèse peut être un des outils utiles à la création de cette couche sémantique intermédiaire.

8.9 Travaux Futurs

8.9.1 Intégration de Fonctionalités de Gestion de Données Images

Une spécification d'un module de gestion de données images est proposée dans [Hudelot, 2005]. Une fonctionalité clé de ce module est la fonctionalité de groupement. Un point de vue intéressant sur le problème de groupement est donné dans [Zlatoff et al., 2004]. Les auteurs de ce travail pensent que le groupement doit tenir compte de la nature des données traitées sans en être complètement dépendant. Le groupement est définit comme le processus qui organise les données images en des structures de haut niveau. Ce processus est naturel pour les humains et a été l'objet de recherches intensives par l'école de psychologie Gestalt. L'intégration de cette fonctionalité serait utile pour grouper les régions résultant d'un processus de segmentation.

8.9.2 Gestion de Différents Types de Données Images

Un élément additionel qui devrait être ajouté à l'approche proposée est la gestion de différents types de données images. Pour le moment, seules des régions sont prises en compte lors de la phase d'apprentissage de concepts visuels et lors de la phase de catégorisation d'objets. Des contours devraient également être pris en compte afin de permettre l'apprentissage et la reconnaissance de structures linéiques (e.g. routes en imagerie aérienne).

8.9.3 Evaluation de l'Utilité de l'Approche pour un Expert en Vision par Ordinateur

Nous avons montré que l'approche apporte des améliorations importantes en terme de convivialité. L'ontologie de concepts visuels permet à des non-spécialistes en vision par ordinateur d'accéder à de complexes algorithmes de vision. Les composants de l'approche proposée (i.e. l'outil d'acquisition de connaissances, le moteur d'apprentissage et de catégorisation) devraient être proposés à un spécialiste en vision par ordinateur. Les questions suivantes devraient alors être posées:

- Est-ce que l'approche proposée est suffisament flexible pour réponde aux besoins du spécialiste en vision par ordinateur?
- Comment le système de catégorisation résultant peut être interfaçé avec d'autres architectures logicielles (e.g. une plate-forme de vidéo-surveillance).
- Dans quelle mesure la productivité de l'expert en vision est améliorée comparé à l'utilisation d'autres librairies de vision ?

8.9.4 Représentation Graphique des Concepts Visuels

Au niveau de l'implémentation, l'outil d'acquisition de connaissances Ontovis devrait être amélioré afin de permettre la visualisation des concepts visuels durant la phase d'acquisition des connaissances. Pour le moment, seuls les labels de concepts visuels sont affichés sous forme textuelle. Un affichage graphique des concepts visuels améliorerait la convivialité de l'outil. Cette perspective est lié à des aspects sémiotiques. La représentation graphique de certains concepts visuels n'est pas évidente. Par exemple, le concept de texture granuleuse a de nombreuses représentations graphiques possibles.

Pour le moment, les concepts visuels peuvent être combinés sous forme de disjonction ou de conjonction. Il serait intéressant d'étudier comment ces concepts visuels pourraient être combinés graphiquement. Cette approche pourrait être comparée au paradigme de requête par esquisse utilisé dans la communauté de la recherche d'images. Un exemple d'une telle approche peut être trouvé dans [Chang et al., 1998] où la notion de requête visuelle patron est introduite.

8.9.5 Amélioration du Calcul de Compatibilité entre Objet Inconnu et Classe

Nous avons vu dans le chapitre 6 que le calcul de la compatibilité entre un objet inconnu et une classe du domaine repose sur une somme pondérée du degré de confiance associé aux concepts visuels reconnus dans l'image et compatibles avec la classe du domaine. Avec cette approche, deux classes ayant la même description visuelle et compatibles avec un objet inconnu posent un problème d'ambiguité. La notion d'importance associée à un attribut peut être utilisée pour résoudre partiellement ce problème. Une autre possibilité est d'introduire un nouveau concept visuel dans l'ontologie afin d'obtenir deux descriptions visuelles différentes. L'importance des attributs n'est pas toujours simple à définir. Des techniques d'optimisation pourraient être utilisées pour maximiser la séparabilité des classes. Ce processus d'optimisation utiliserait des images annotées en terme de concepts visuels et en terme de classes du domaine.

8.9.6 Utilisation de Maillages 3-D Annotés Sémantiquement

Nous avons vu en section 5.2.2 que des maillages 3-D peuvent être utilisés pour obtenir des projections 2-D des objets d'intérêts. Les projections 2-D résultantes sont alors labellisées par des concepts visuels. Pour les moment, les maillages 3-D sont projetés entièrement afin d'obtenir les projections 2-D. Des labels sémantiques sont souvent associés aux primitives composant les modèles 3-D (e.g le label *roue* associé au *cylindre* utilisé pour modéliser la roue d'une voiture). Il serait interéssant d'utiliser ces labels pour obtenir des projections de sous-parties spécifiques des objets d'intérêts. L'annotation des sous-parties des objets serait ainsi simplifiée.

8.9.7 Utilisation de Descripteurs Bas Niveau pour l'Apprentissage et la Détection de Relations Spatiales Complexes

Le formalisme introduit au chaptire 4 permet la description de relations binaires entre objets. Ce point doit être amélioré par l'introduction de relations n-aire. Ceci implique que des descripteurs bas niveau seront nécessaires pour détecter ces relations. Comme expliqué dans [Scott et al., 2005], l'utilisation des histogrammes de Matsakis [Matsakis et al., 2004] sont utiles pour caractériser les relations spatiales entre plusieurs objets de manière invariante aux changements d'échelles, aux rotations et aux translations des configurations spatiales.

8.9.8 Amélioration de la Phase de Segmentation Manuelle

Dans le chapitre 5, l'intérêt d'une segmentation semi-automatique a été souligné. En particulier, la technique appellée *ciseaux intelligents* a montré son efficacité pour la segmentation précise d'objets. Une nouvelle technique appellée *Lazy Snapping* [Li et al., 2004] semble encore plus efficace. Cette technique sépare les traitements fins des traitements grossiers, permettant ainsi de facilement désigner des objets et d'ajuster le résultat en détail. Des études de faisabilité ont montré que cette approche est plus satisfaisante du point de vue de l'utilisateur et qu'elle produit de meilleurs résultats que d'autres techniques avancées de segmentation assistée. Une autre technique appellée Grabcut [Rother et al., 2004] donne aussi d'excellents résultats tout en demandant peu d'efforts de l'utilisateurs: il suffit de tracer un rectangle autour de l'objet d'intérêt pour obtenir ensuite une segmentation précise.

8.9.9 Vers une Segmentation Adaptative

Le chapitre 7 a montré l'importance de la qualité de la segmentation. De nombreux progrès doivent être faits à ce niveau. Un travail actuellement mené dans l'équipe Orion par V.Martin [Martin et al., 2006] traite du problème de la segmentation adaptative. Dans ce travail, un schéma est proposé pour sélectionner automatiquement des algorithmes de segmentation et régler leurs paramètres. Cette approche repose sur des images segmentées manuellement. La phase d'apprentissage repose sur des techniques d'apprentissage artificiel et est composée de deux étapes : calcul des paramètres optimaux par optimisation et apprentissage de la sélection d'algorithme. A la fin de la phase d'apprentissage, toutes les images segmentées manuellement sont associées à un ensemble de paramètres optimaux et un algorithme de segmentation. Cet ensemble forme une base de case. La phase de segmentation automatique utilise cette base de cas pour permettre une segmentation adaptative (fig. 8.7). Les descripteurs bas niveau extraits de l'image sont donnés en entrée de l'algorithme de sélection entrainé durant la phase d'apprentissage (1). Ensuite, le cas le plus proche contenu dans la base de cas est sélectionné. Une fois le cas le proche sélectionné, l'image est segmentée avec l'algorithme et les paramètres associés au cas. Cette approche a donné des résultats prometteurs et devrait être integrée avec les travaux présentés dans ce manuscrit.

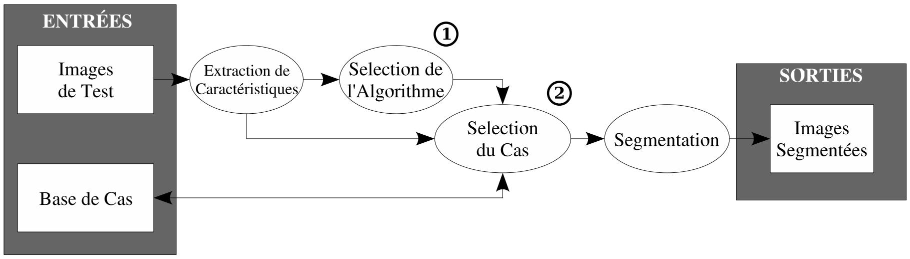


Figure 8.7: Vers une segmentation d'images adaptive. Après une phase d'apprentissage utilisant des images segmentées manuellement, les images de test sont segmentées de manière adaptative (i.e. parametrisation automatique et sélection d'algorithme).

8.10 Perspectives à Long Terme

8.10.1 Coopération avec VSIP

La plate-forme de vidéo surveillance développée au sein de l'équipe Orion [Avanzi et al., 2005] est utilisée pour l'analsyse du comportement de nombreux types d'objets mobiles (e.g. voitures, personnes). Pour le moment, la reconnaissance des différentes catégories d'objets repose sur des descripteurs bas-niveau comme la *hauteur* ou la *largeur* des objets mobiles. Ceci pourrait être amélioré en interfaçant cette plate-forme de vidéo-surveillance avec notre plate-forme de catégorisation d'objets. Une première phase consisterait à utiliser l'outil d'acquisition de connaissances, Ontovis, afin de définir les objets impliqués dans les scénarios à reconnaitre. Des images d'exemple de ces objets devraient aussi être fournies. L'interaction avec la plate-forme de vidéo surveillance se ferait par le biais de requêtes et de résultats de catégorisation (fig. 8.8).

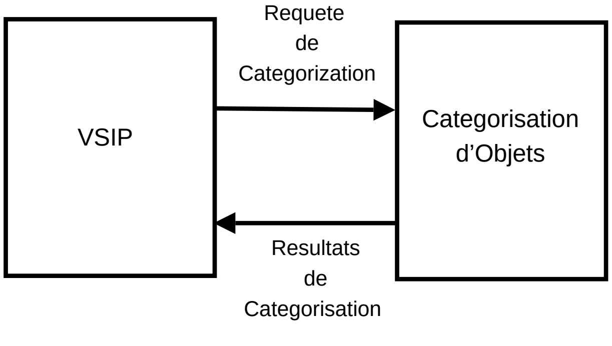


Figure 8.8: La plate-forme de vidéo surveillance développée au sein de l'équipe Orion (VSIP) pourrait fonctionner en coopération avec la plate-forme de catégorisation d'objets.

Cette approche donnerait un système doté des fonctionalités suivantes: détection et localisation, suivi, classification et catégorisation, formation de concept et visualisation.

8.10.2 Combinaison du Paradigme de Requête par l'Exemple avec celui de Requête par Concept

Dans le chapitre 7, nous avons montré comment l'approche proposée a été utilisée pour des besoins d'indexation et de recherche d'images. L'indexation et la recherche se font dans les termes du domaine ainsi qu'en terme de concepts visuels. C'est une approche qui appartient au paradigme de la recherche par concept. Il serait possible de combiner cette approche avec une approche appartenant au paradigme de la recherche par l'exemple. Comme expliqué dans [Town and Sinclair, 2004], la recherche par exemple ne permet pas de capturer l'essence conceptuelle des images utilisées comme requêtes. Notre approche pourrait être utilisée pour dériver une description symbolique des images requêtes afin de produire une requête symbolique. Ceci consisterait à transformer une image (ou une sous-partie de cette image) requête en concepts visuels. Ainsi, la recherche reposerait sur la description symbolique de la requête image (fig. 8.9).

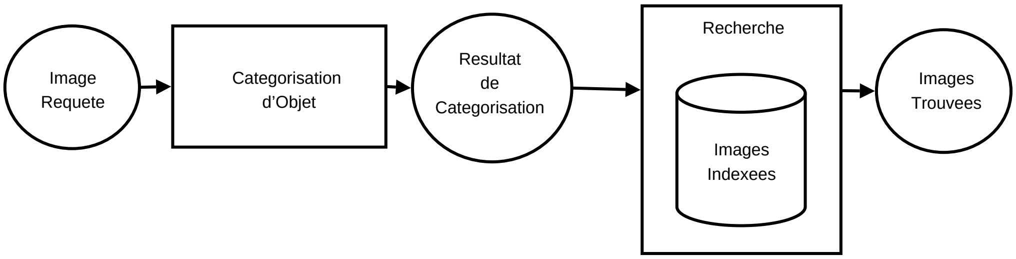


Figure 8.9: Combinaison du paradigme de la recherche par l'exemple avec celui de la recherche par concept. La requête image est utilisée comme entrée du processus de catégorisation. Le résultat de catégorisation est utilisé pour la recherche dans la base d'images indexées.

8.10.3 Amélioration du Processus d'Annotation

Nous avons vu dans le chapitre 5 que l'annotation et la segmentation des images d'exemple se fait au niveau des concepts visuels. Le résultat de ce processus est un ensemble de régions annotées par des concepts visuels, et non par des classes du domaine. Il serait plus convivial de pouvoir annoter ces régions par des classes du domaine. Le nombre d'annotations nécessaires serait ainsi réduit. Le compromis suivant pourrait covenir: une partie des images d'exemple pourraient être annotées par de concepts visuels *et* des classes du domaines; l'autre partie de la bases d'exemple ne serait que partiellement annotées par des classes du domaine. Les annotations complètes seraient alors obtenues par analogie avec les images complètement annotées.

8.10.4 Retour de Pertinence

Un élément important qui pourrait être apporté à l'approche proposée est l'introduction de mécanismes de *retour de pertinence*. Nous avons vu qu'un résultat de catégorisation est éxprimé sous la forme de concepts visuels détectés et compatibles avec une classe du domaine. Deux types d'erreurs peuvent être constatés: des faux positifs et des faux négatifs aussi bien au niveau des concepts visuels qu'au niveau des classes du domaine. Ainsi, le retour de pertinence pourrait s'effectuer à ces deux niveaux de manière interactive. L'utilisateur pourrait ainsi:

- Ajouter au résultat de catégorisation des concepts visuels non détectés dans l'image (i.e. faux négatifs).
- Enlever des concepts visuels détectés dans l'image (i.e. faux positifs).
- Modifier le nom des classes compatibles avec l'objet inconnu.

8.10.5 Bases de Connaissances Dynamiques

Le processus de catégorisation est guidé par la connaissance acquise. Un objet dans une image ne peut être reconnu que comme du type d'une classe de la taxonomie. C'est la limite de l'hypothèse de *monde fermé* de la plupart des systèmes à base de connaissances. Dans le chapitre 6, la notion de rejet en distance a été présentée. Dans certains cas, plusieurs types de concepts visuels peuvent être rejetés simultanément. Il serait intéressant d'étudier statistiquement la co-occurence des concepts visuels rejetés. Des configurations spécifiques de concepts visuels rejetés peuvent correspondre à de nouvelles classes du domaine. Les régions correspondant aux concepts visuels rejetés peuvent alors être proposés à l'expert afin de les annoter.

8.10.6 Apprentissage Incrémental

Pour le moment, lorsqu'un nouvel exemple est ajouté à l'ensemble des régions annotées, le processus d'apprentissage décrit dans le chapitre 5 doit s'effectuer à nouveau complètement. Le processus d'optimisation utilisé dans le cadre des Machines à Vecteurs de Support tient compte de tous les exemples. L'utilisation de techniques d'apprentissage incrémental permettrait d'obtenir la propriété d'apprentissage en continu. Un modèle d'apprentissage Bayésien comme celui utilisé dans [Fei-Fei et al., 2004] semble adapté à ce problème.

8.10.7 Analyse et Recherche de Vidéos

La présentation invitée de A.Hauptmann [Hauptmann, 2005] à CIVR 2005 a montré qu'il reste beaucoup à faire dans le domaine de l'analyse et de la recherche de vidéos. Concernant les extensions possibles de cette thèse, l'ajout de concept visuels temporels est un point particulièrement important. Ces concepts pourraient permettre la description du mouvement des objets d'intérêt. Par exemple, la description de la trajectoire d'une voiture peut se faire à l'aide de concepts géométriques. Des concepts temporels pour décrire l'évolution morphologique des objets devraient également être ajoutés à l'ontologie. Dans le cas de vidéos génériques, le problème de l'estimation de la représentation de trajectoires dans un environnement non calibré devra être traité [Nunziati et al., 2005].

8.10.8 Indexation et Recherche dans des Bases d'Images Génériques

Dans le cadre de la recherche et de l'indexation d'images, l'approche proposée est particulièrement adaptée à des domaines d'application dédiés (e.g dans les domaine des véhicules de transport). La plupart des concepts de l'ontologie sont valides pour différents types d'applications (e.g. concepts de teinte). L'indexation de bases d'images génériques pourrait se faire en entrainant ces détecteurs avec des images de différentes applications. La base d'images ne serait alors indexée qu'en terme de concepts visuels. La recherche ne pourrait alors se faire qu'en terme de concepts visuels. Les requêtes seraient alors d'un niveau sémantique intermédiaire. Ceci serait utile pour la recherche d'images ayant une apparence visuelle bien caractéristique

8.10.9 Meilleure Utilisation des Concepts Visuels pour Guider la Segmentation

Dans le chapitre 6, nous avons vu que les concepts de position et de taille guident le processus de segmentation automatique. Un travail doit être mené afin de tenir compte d'autres catégories de concepts visuels. Ceci pourrait être fait par une approche à base de pilotage de programmes [Thonnat et al., 1999] ou encore par une approche à base de règles.

8.10.10 Utilisation de Geons pour la Modélisation des Objets

La notion de geon a été introduit par Biederman dans [Bierderman, 1987]. Cette notion s'est ensuite propagé dans la communauté de la vision par ordinateur. Les geons sont des primitives géométriques qui peuvent être utilisées afin de modéliser de nombreux types d'objets. Le potentiel de geons pour la reconnaissance d'objets est discuté dans [Dickinson et al., 1997]. Les geons ont été utilisé pour des besoins de reconnaissance d'objets 3-D dans [Borges and Fisher, 1996]. Une méthode pour approximer la forme d'objets 3-D en utilisant des geons paramétriques est présentée dans [Wu and Levine, 1997]. Les geons pourraient s'intégrer dans l'ontologie de concepts visuels pour permettre aux experts de modéliser des objets 3-D complexes. L'extraction de geons à partir d'images 2-D est encore un problème ouvert.

8.10.11 Application à des Images de Profondeur

Des images de profondeur sont obtenues par le biais de capteurs de stéréo vision. Le résultat d'acquisition est une carte de profondeur. Des concepts visuels 3-D dédiés à la description et à la reconnaissance de l'apparence d'objets visualisés dans des images de profondeur serait utile (e.g dans les applications robotiques). Une vue d'ensemble de ce sujet de recherche peut être trouvée dans [Arman and Aggarwal, 1993]. Un travail important devra être mené au niveau de l'extraction de descripteurs numériques ainsi qu'au niveau de la segmentation automatique. Des relations spatiales 3-D devront aussi être introduites afin de décrire des relations entre objets dans un monde 3-D. Le principe d'utilisation des geons est donné en fig. 8.10.

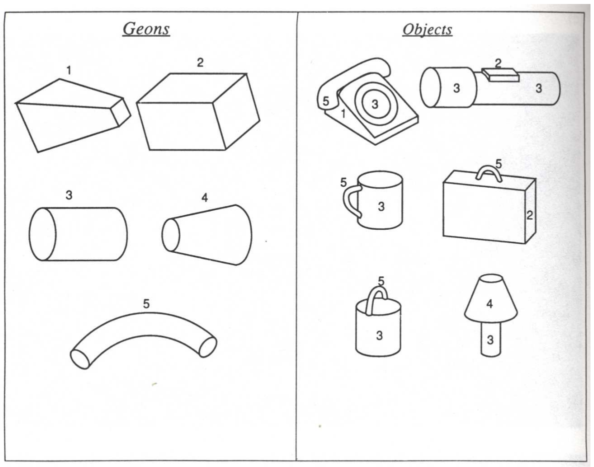


Figure 8.10: Les geons de Biederman [Bierderman, 1987] et leur utilisation pour la modélisation d'objets.

Bibliography

[Arman and Aggarwal, 1993] Arman, F. and Aggarwal, J. K. (1993). Model-based object recognition in dense-range images-a review. *ACM Comput. Surv.*, 25(1):5–43.

[Auer et al., 2005] Auer, P., Bloch, I., Buxton, H., Courtney, P., Dickinson, S., Fisher, B., Granlund, G., Neumann, B., Pinz, A., Sandini, G., Sommer, G., Vernon, D., Billard, A., Boettcher, P., Christensen, H., Crooker, A., Erbest, C., Förster, W., Hlavac, V., Leonardis, A., Nagel, H.-H., Niemann, H., Pirri, F., Schiele, B., Tsotsos, J., Vincze, M., Bischof, H., Bülthoff, H., Cohn, T., Crowley, J., Eklundh, J.-O., Gilby, J., Kittler, J., Little, J., Nebel, B., Paletta, L., Sagerer, G., Simpson, R., and Thonnat, M. (2005). A research roadmap of cognitive vision. Technical report.

[Avanzi et al., 2005] Avanzi, A., Brémond, F., Tornieri, C., and Tonnat, M. (2005). Design and assessment of an intlligent activity monitoring platform. *EURASIP*, 14:2359–2374.

[Ayache and Faugeras, 1986] Ayache, N. and Faugeras, O. D. (1986). Hyper: a new approach for the recognition and positioning to two-dimensional objects. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(1):44–54.

[Baader et al., 2003] Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F., editors (2003). *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press.

[Bachimont, 2000] Bachimont, B. (2000). *Engagement semantique et engagement ontologique : conception et realisation d'ontologie en ingenierie des connaissances dans* ***Ingeniere des connaissances, evolutions recentes et nouveaux defis***. Eyrolles.

[Barthes, 1977] Barthes, R. (1977). *Image, Music, Text.* Fotana.

[Basri, 1996] Basri, R. (1996). Recognition by prototypes. *Int. J. Comput. Vision*, 19(2):147–167.

[Belkhatir et al., 2004] Belkhatir, M., Mulhem, P., and Chiaramella, Y. (2004). Integrating perceptual signal features within a multi-facetted conceptual model for automatic image retrieval. In *ECIR*, pages 267–282.

[Belongie et al., 2001] Belongie, S., Malik, J., and Puzicha, J. (2001). Matching shapes. In *ICCV*, pages 454–463.

[Bhushan et al., 1997] Bhushan, N., Rao, A., and Lohse, G. (1997). The texture lexicon: Understanding the categorization of visual texture terms and their relationship to texture images. *Cognitive Science*, 21(1):219–246.

[Bierderman, 1987] Bierderman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147.

[Blazquez et al., 1998] Blazquez, M., Fernandez, M., Garcia-Pinar, J., and Gómez-Pérez, A. (1998). Building ontologies at the knowledge level using the ontology design environment. In *KAW98*.

[Borges and Fisher, 1996] Borges, D. L. and Fisher, R. B. (1996). Class-based recognition of 3d objects represented by volumetric primitives. In *BMVC*.

[Borgida et al., 1989] Borgida, A., Brachman, R. J., McGuinness, D. L., and Resnick, L. A. (1989). Classic: a structural data model for objects. In *SIGMOD '89: Proceedings of the 1989 ACM SIGMOD international conference on Management of data*, pages 58–67, New York, NY, USA. ACM Press.

[Boujemaa and Fauqueur, 2003] Boujemaa, N. and Fauqueur, J. (2003). What's beyond query by example? Technical Report 5068, INRIA.

[Brigger et al., 1998] Brigger, P., Engel, R., and Unser, M. (1998). B-spline snakes and a java interface: An intuitive tool for general contour outlining. In *ICIP (2)*, pages 277–281.

[Brodatz, 1966] Brodatz, P. (1966). *Textures: A Photographic Album for Artists and Designers*. Dover Publications, New York.

[Brooks, 1983] Brooks, R. A. (1983). Model-based three-dimensional interpretations of two-dimensional images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 5(2):140–150.

[Burges, 1998] Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

[Campbell and Krumm, 2000] Campbell, R. and Krumm, J. (2000). Object recognition for an intelligent room. In *CVPR*, pages 1691–1697.

[Carson et al., 1999] Carson, C., Thomas, M., Belongie, S., Hellerstein, J. M., and Malik, J. (1999). Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*. Springer.

[Chang et al., 1998] Chang, S., Chen, W., and Sundaram, H. (1998). Semantic visual templates: Linking visual features to semantics. pages 531–535.

[Chen et al., 2004] Chen, J., Pappas, T. N., Mojsilivic, A., and Rogowitz, B. E. (2004). Adaptive perceptual color-texture image segmentation. *Transactions on Image Processing*, 14(10).

[Christel and Hauptmann, 2005] Christel, M. and Hauptmann, A. (2005). The use and utility of high-level semantic features in video retrieval. In Springer, editor, *Proc. of International Conference on Image and Video Retrieval*, volume 3568 of *Lecture Notes in Computer Science*, pages 135–144.

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

[Clementini et al., 1993] Clementini, E., Felice, P. D., and van Oosterom, P. (1993). A small set of formal topological relationships suitable for end-user interaction. In *SSD '93: Proceedings of the Third International Symposium on Advances in Spatial Databases*, pages 277–295, London, UK. Springer-Verlag.

[Clouard et al., 1999] Clouard, R., Elmoataz, A., Porquet, C., and Revenu, M. (1999). Borg: A knowledge-based system for automatic generation of image processing programs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(2):128–144.

[Coenen and Visser, 1999] Coenen, F. and Visser, P. (1999). A generic ontology for spatial reasoning. In Miles, R., Moulton, M., and Bramer, M., editors, *Proceedings of ES98, the Eighteenth Annual International Conference of the British Computer Society Specialist Group on Expert Systems, Cambridge UK, December 14th-16th 1998*, Research and Development in Expert Systems XV, page 14. Springer Verlag.

[Cohn and Hazarika, 2001] Cohn, A. G. and Hazarika, S. M. (2001). Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 46(1-2):1–29.

[Cohn et al., 2003] Cohn, A. G., Magee, D., Galata, A., Hogg, D., and Hazarika, S. (2003). Towards an architecture for cognitive vision using qualitative spatio-temporal representations and abduction. In C. Freksa, W. Brauer, C. H. and Wender, K. F., editors, *Spatial Cognition III, Routes and Navigation, Human Memory and Learning, Spatial Representation and Spatial Learning*, pages 232–248. Springer-Verlag.

[Comaniciu and Meer, 2002] Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619.

[Corcho and Gómez-Pérez, 2000] Corcho, O. and Gómez-Pérez, A. (2000). A roadmap to ontology specification languages. In *Knowledge engineering and knowledge management methods,models and tools. EKAW 2000*, pages 80–96.

[Crevier and Lepage, 1997] Crevier, D. and Lepage, R. (1997). Knowledge-based image understanding systems: A survey. *Computer Vision and Image Understanding*, 67(2):161—185.

[Csurka et al., 2004] Csurka, G., Dance, C., Bray, C., Fan, L., and Willamowski, J. (2004). Visual categorization with bags of keypoints. In *Pattern Recognition and Machine Learning in Computer Vision Workshop*, Grenoble, France.

[Deriche, 1990] Deriche, R. (1990). Fast algorithms for low-level vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(1):78–87.

[Dey et al., 2002] Dey, N., Boucher, A., and Thonnat, M. (2002). Image formation model of 3-d tranlucent object observed in light microscopy. In *Proceedings of ICIP'02*, volume 2, pages 469–472.

[Dickinson et al., 1997] Dickinson, S., Bergevin, R., Biederman, I., Eklundh, J., Munck-Fairwood, R., Jain, A., and Pentland, A. (1997). Panel report: The potential of geons for generic 3-d object recognition. *IVC*, 15(4):277–292.

[Dijkstra, 1959] Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271.

[Draper et al., 1989] Draper, B., Collins, R., Brolio, J., Hanson, A., and Riseman, E. (1989). The schema system. *The International Journal of Computer Vision*, 2(3):209–250.

[Draper et al., 1996] Draper, B., Hanson, A., and Riseman, E. (1996). Knowledge-directed vision: control, learning and integration.

[Dubuisson and Masson, 1993] Dubuisson, B. and Masson, M. (1993). A statistical decision rule with incomplete knowledge about classes. *Pattern Recognition*, 26(1):155–165.

[Dumitras and Venetsanopoulos, 2001] Dumitras, A. and Venetsanopoulos, A. N. (2001). A comparative study of snake models with application to object shape description in bi-level and gray-level images. In *Proceedings of IEEE-Eurasip Workshop on Nonlinear Signal and Image Processing*, Boston, USA.

[Edelman, 1997] Edelman, S. (1997). Computational theories of object recognition. *Trends in Cognitive Sciences*, 1:296–304.

[Fauqueur and Boujemaa, 2003] Fauqueur, J. and Boujemaa, N. (2003). New image retrieval paradigm: logical composition of region categories. In *ICIP03*, pages III: 601–604.

[Fei-Fei et al., 2004] Fei-Fei, L., Fergus, R., and Perona., P. (2004). Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *CVPR 2004, Workshop on Generative-Model Based Vision.*

[Fergus et al., 2003] Fergus, R., Perona, P., and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *IEEE Conf. Computer Vision and Pattern Recognition, 2003.*

[Gandon, 2002] Gandon, F. (2002). Ontology engineering: A survey and a return on experience. Technical Report 4396, INRIA.

[Grimson and Huttenlocher, 1990] Grimson, W. E. L. and Huttenlocher, D. P. (1990). On the sensitivity of the hough transform for object recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(3):255–274.

[Gruber, 1993] Gruber, T. R. (1993). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In Guarino, N. and Poli, R., editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands. Kluwer Academic Publishers.

[Guarino and Giaretta, 1995] Guarino, N. and Giaretta, P. (1995). Ontologies and knowledge bases: Towards a terminological clarification. In Mars, N., editor, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing Amsterdam, NL. IOS Press*, pages 25–32.

[Haarslev and Möller, 2001] Haarslev, V. and Möller, R. (2001). Description of the RACER system and its applications. In *Description Logics*.

[Haralick, 1979] Haralick, R. M. (1979). Statistical and structural approaches to texture. *Proceedings IEEE*, 67(5):786–804.

[Harnad, 1990] Harnad, S. (1990). The symbol grounding problem. *Physica*, D(42):335–346.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Proceedings of the Fourth Alvey Vision Conference (Manchester University, 31st August–2nd September)*, pages 147–152. The University of Sheffield Printing Unit.

[Hauptmann, 2005] Hauptmann, A. (2005). Lessons for the future from a decade of informedia video analysis research. In Springer, editor, *Proc. of International Conference on Image and Video Retrieval*, volume 3568 of *Lecture Notes in Computer Science*, pages 1–10.

[Hauptmann, 2004] Hauptmann, A. G. (2004). Towards a large scale concept ontology for broadcast video. In *CIVR*, pages 674–675.

[Havaldar et al., 1996] Havaldar, P., Medioni, G., and Stein, F. (1996). Perceptual grouping for generic recognition. *IJCV*, 20(1/2):59–80.

[Horrocks, 1998] Horrocks, I. (1998). The FaCT system. In de Swart, H., editor, *Automated Reasoning with Analytic Tableaux and Related Methods: International Conference Tableaux'98*, number 1397 in Lecture Notes in Artificial Intelligence, pages 307–312. Springer-Verlag.

[Hudelot, 2005] Hudelot, C. (2005). *Towards a Cognitive Vision Platform for Semantic Image Interpretation; Application to the Recognition of of Biological Organisms.* PhD thesis, Université de Nice-Sophia Antipolis.

[Jurie and Schmid, 2004] Jurie, F. and Schmid, C. (2004). Scale-invariant shape features for recognition of object categories. In *CVPR (2)*, pages 90–96.

[Leibe, 2004] Leibe, B. (2004). *Interleaved Object Categorization and Segmentation.* PhD thesis, ETH Zurich.

[Leibe et al., 2004] Leibe, B., Leonardis, A., and Schiele, B. (2004). Combined object categorization and segmentation with an implicit shape model. In *Proceedings of the Workshop on Statistical Learning in Computer Vision*, Prague, Czech Republic.

[Li and Wang, 2003] Li, J. and Wang, J. Z. (2003). Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1075–1088.

[Li et al., 2004] Li, Y., Sun, J., Tang, C.-K., and Shum, H.-Y. (2004). Lazy snapping. *ACM Trans. Graph.*, 23(3):303–308.

[Liu et al., 1994] Liu, S., Thonnat, M., and Berthod, M. (1994). Automatic classification of planktonic foraminifera by a knowledge-based system. In *The Tenth Conference on Artificial Intelligence for Applications*, pages 358–364, San Antonio, Texas. IEEE Computer Society Press.

[Lowe, 1999] Lowe, D. G. (1999). Object Recognition From Local Scale-Invariant Features. In *International Conference on Computer Vision (ICCV)*, pages 1150–1157.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110.

[MacGregor, 1991] MacGregor, R. (1991). Inside the loom description classifier. *SIGART Bulletin*, 3(2):88–92.

[Manjunath and Ma, 1996] Manjunath, B. and Ma, W. (1996). Texture features for browsing and retrieval of image data. *PAMI*, 18(8):837–842.

[Marr, 1982] Marr, D. (1982). *Vision: a computational investigation into the human representation and processing of visual information.* W. H. Freeman, San Francisco.

[Martin et al., 2006] Martin, V., Maillot, N., and Thonnat, M. (2006). Learning for adaptive image segmentation. In *ICVS, New York, USA*. IEEE.

[Matsakis et al., 2004] Matsakis, P., Keller, J. M., Sjahputera, O., and Marjamaa, J. (2004). The use of force histograms for affine-invariant relative position description. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(1):1–18.

[Matsuyama and Hwang, 1990] Matsuyama, T. and Hwang, V.-S. (1990). *SIGMA - A Knowledge-Based Aerial Image Understanding System*. Plenum Press New York USA.

[McGuinness and van Harmelen, 2004] McGuinness, D. and van Harmelen, F. (2004). Owl web ontology language overview. Technical report, W3C Recommendation.

[Medioni and François, 2000] Medioni, G. G. and François, A. R. J. (2000). 3-d structures for generic object recognition. In *ICPR*, pages 1030–1037.

[Mezaris et al., 2004] Mezaris, V., Kompatsiaris, I., , and Strintzis, M. (2004). Region-based image retrieval using an object ontology and relevance feedback. *EURASIP JASP*, 2004(6):886–901.

[Mikolajczyk and Schmid, 2004] Mikolajczyk, K. and Schmid, C. (2004). Scale & affine invariant interest point detectors. *Int. J. Comput. Vision*, 60(1):63–86.

[Miller and Johnson-Laird, 1976] Miller, G. and Johnson-Laird, P. (1976). *Language and Perception*. Cambridge University Press.

[Minsky, 1975] Minsky, M. (1975). A framework for representing knowledge. *The Psychology of Computer Vision*, pages 211–277.

[Moisan, 1998] Moisan, S. (1998). *Une plate-forme pour une programmation par composants de systèmes à base de connaissances*. Habilitation à diriger les recherches, Université de Nice.

[Moisan et al., 2001] Moisan, S., Ressouche, A., and Rigault, J.-P. (2001). Blocks, a component framework with checking facilities for knowledge-based systems. *Informatica, Special Issue on Component Based Software Development*, 25(4):501–507.

[Möller et al., 1999] Möller, R., Neumann, B., and Wessel, M. (1999). Towards computer vision with description logics: Some recent progress. In *Proceedings Integration of Speech and Image Understanding, Corfu, Greece*, pages 101–115.

[Mortensen and Barrett, 1998] Mortensen, E. N. and Barrett, W. A. (1998). Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing*, 60(5):349–384.

[Mortensen and Barrett, 1999] Mortensen, E. N. and Barrett, W. A. (1999). Toboggan-based intelligent scissors with a four-parameter edge model. In *CVPR*, pages 2452–2458.

[Mundy and Zisserman, 1992] Mundy, J. L. and Zisserman, A., editors (1992). *Geometric invariance in computer vision*. MIT Press, Cambridge, MA, USA.

[Nazif and Levine, 1986] Nazif, A. and Levine, M. (1986). Low level image segmentation: An expert system. *PAMI*, 8(5):676.

[Neumann and Weiss, 2003] Neumann, B. and Weiss, T. (2003). Navigating through logic-based scene models for high-level scene interpretations. In Crowley, J. L., Piater, J. H., Vincze, M., and Paletta, L., editors, *Computer Vision Systems, Third International Conference, ICVS*, volume 2626 of *Lecture Notes in Computer Science*. Springer.

[Niemann et al., 1990] Niemann, H., Sagerer, G., Schröder, S., and Kummert, F. (1990). Ernest: A semantic network system for pattern understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(9):883–905.

[Nunziati et al., 2005] Nunziati, W., Sclaroff, S., and Bimbo, A. D. (2005). An invariant representation for matching trajectories across uncalibrated video streams. In *CIVR*, pages 318–327.

[Ossola et al., 1996] Ossola, J., Brémond, F., and Thonnat, M. (1996). A communication level in a distributed architecture for object recognition. In *8th International Conference on Systems Research Informatics and Cybernetics*.

[Pass et al., 1996] Pass, G., Zabih, R., and Miller, J. (1996). Comparing images using color coherence vectors. In *ACM Multimedia*, pages 65–73.

[Picard, 1995] Picard, R. W. (1995). Toward a visual thesaurus. In *MIRO*.

[Platt, 2000] Platt, J. (2000). Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. pages 61–74.

[Precioso et al., 2005] Precioso, F., Barlaud, M., Blu, T., and Unser, M. (2005). Robust real-time segmentation of images and videos using a smooth-spline snake-based algorithm. *IEEE Transactions on Image Processing*, 14(7):910–924.

[Quillian, 1985] Quillian, M. R. (1985). Word concepts: A theory and simulation of some basic semantic capabilities. In Brachman, R. J. and Levesque, H. J., editors, *Readings in Knowledge Representation*, pages 97–118. Kaufmann, Los Altos, CA.

[Rao and Lohse, 1993] Rao, A. and Lohse, G. (1993). Towards a texture naming system: Identifying relevent dimensions of texture. *Visual Research*, 36(11):1649–1669.

[Reed and Wechsler, 1990] Reed, T. R. and Wechsler, H. (1990). Segmentation of textured images and gestalt organization using spatial/spatial-frequency representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(1):1–12.

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004). ”grabcut”: interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314.

[Sangineto, 2003] Sangineto, E. (2003). An abstract representation of geometric knowledge for object classification. *Pattern Recogn. Lett.*, 24(9-10):1241–1250.

[Schiele and Crowley, 2000] Schiele, B. and Crowley, J. L. (2000). Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50.

[Schmid and Mohr, 1997] Schmid, C. and Mohr, R. (1997). Local greyvalue invariants for image retrieval. *Pattern Analysis and Machine Intelligence*, 19(5).

[Scholkopf and Smola, 2001] Scholkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.

[Sciascio et al., 2002] Sciascio, E., M.Donini, F., and Mongiello., M. (2002). Structured knowledge representation for image retrieval. *Journal of Artificial Intelligence Research*, 16:209–257.

[Scott et al., 2005] Scott, G., Klaric, M., and Shyu, C. (2005). Modeling multi-object spatial relationships for satellite image database indexing and retrieval. In Springer, editor, *Proc. of International Conference on Image and Video Retrieval*, volume 3568 of *Lecture Notes in Computer Science*, pages 247–256.

[Shapira and Ullman, 1991] Shapira, Y. and Ullman, S. (1991). A pictorial approach to object classification. In *IJCAI*, pages 1257–1263.

[Shatford, 1986] Shatford, S. (1986). Analyzing the subject of a picture : a theoretical approach. *Cataloguing and Classification Quaterly*, (6):39–62.

[Shekhar et al., 1998] Shekhar, C., Moisan, S., Vincent, R., Burlina, P., and Chellappa, R. (1998). Knowledge-based control of vision systems. *Image and Vision Computing*, 17:667–683.

[Soo et al., 2003] Soo, V.-W., Lee, C.-Y., Li, C.-C., Chen, S. L., and Chen, C.-C. (2003). Automated semantic annotation and retrieval based on sharable ontology and case-based learning techniques. In *JCDL '03*, pages 61–72. IEEE Computer Society.

[Sowa, 1984] Sowa, J. F. (1984). *Conceptual structures: information processing in mind and machine*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Swain and Ballard, 1991] Swain, M. J. and Ballard, D. H. (1991). Color indexing. *Int. J. Comput. Vision*, 7(1):11–32.

[Thonnat and Bijaoui, 1989] Thonnat, M. and Bijaoui, A. (1989). Knowledge-based galaxy classification systems. *Knowledge-based systems in astronomy, Lecture Notes in Physics.*, 329.

[Thonnat et al., 1999] Thonnat, M., Moisan, S., and Crubézy, M. (1999). Experience in integrating image processing programs. In Henrik Christensen, editor, *Proceeding of the*

1st International Conference on Vision Systems, Lecture Notes in Computer Science, Las Palmas, Gran Canaria, Spain. Springer-Verlag.

[Town and Sinclair, 2004] Town, C. and Sinclair, D. (2004). Language-based querying of image collections on the basis of an extensible ontology. *IVC*, 22(3):251–267.

[Ullman and Basri, 1991] Ullman, S. and Basri, R. (1991). Recognition by linear combinations of models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(10):992–1006.

[Vapnik, 1995] Vapnik, V. N. (1995). *The nature of statistical learning theory.* Springer-Verlag New York, Inc., New York, NY, USA.

[Vu et al., 2003] Vu, V.-T., Brémond, F., and Thonnat, M. (2003). Automatic video interpretation: A novel algorithm for temporal scenario recognition. In *IJCAI*, pages 1295–1302.

[Weber et al., 2000] Weber, M., Welling, M., and Perona, P. (2000). Towards automatic discovery of object categories. In *CVPR*.

[Wu and Levine, 1997] Wu, K. and Levine, M. D. (1997). 3-d shape approximation using parametric geons. *Image Vision Comput.*, 15(2):143–158.

[Zhang and Tan, 2002] Zhang, J. and Tan, T. (2002). Brief review of invariant texture analysis methods. *Pattern Recognition*, 35(3):735–747.

[Zlatoff et al., 2004] Zlatoff, N., Tellez, B., and Baskurt, A. (2004). Image understanding and scene models: a generic framework integrating domain knowledge and gestalt theory. In *ICIP*, pages 2355–2358.