

A Meta-Learning Approach to Ground Symbols from Visual Percepts

Nicolas Bredeche^{a,b} Yann Chevaleyre^a Jean-Daniel Zucker^{a,1}
Alexis Drogoul^a Gérard Sabah^b

^a*LIP6-CNRS, Boite 169, Université P&M Curie
4, Place Jussieu, 75232 PARIS Cedex 6, France*

^b*LIMSI-CNRS, Université Paris XI
BP 133, F-91403 ORSAY CEDEX, France*

Abstract

There is a growing interest in both the robotics and AI communities to give autonomous robots the ability to interact with humans. To efficiently identify properties from its environment (be it the presence of a human, or a fire extinguisher or another robot of its kind) is one of the critical task for supporting meaningful Robot/Human dialogues. This task is a particular anchoring task. Our goal is to endow autonomous mobile robots (in our experiments a Pioneer 2DX) with a perceptual system that can efficiently adapt itself to the context so as to enable the learning task required to physically ground symbols. In effect, Machine learning based approaches to provide robots with an ability to ground symbols heavily rely on ad-hoc perceptual representation provided by AI designers. Our approach is in the line of meta-learning algorithms, that iteratively change representations so as to discover one that is well fitted for the task. The architecture we propose is based on a widely used approach in constructive induction: the Wrapper-model. Experiments using the PLIC system to have a robot identify the presence of Humans and Fire Extinguishers show the interest of such an approach that dynamically abstracts a well fitted image description depending on the concept to learn.

Key words: Anchoring, Meta-Learning, Change of Representation, Object Identification.

Email address: `Nicolas.Bredeche@lip6.fr` (Nicolas Bredeche).

¹ This author is supported by a research grant from the CNRS ("délégation"), and a cognitive project (on spatial cognition).

1 Introduction: Anchoring symbols and identifying objects

Recent works in both Robotics and Artificial Intelligence have shown a growing interest in providing mobile robots with the ability to interact and communicate with humans. One of the main challenges in designing such robots is to give them the ability to perceive the world in a way that is useful or understandable to us. One approach is to give the robot the ability to identify physical entities and relate them to perceptual symbols that are used by humans (to refer to these same physical entities). To perform this task, the robot has to ground these symbols to its percepts (i.e., its sensor data). Recently, the term of *Anchoring*[1] has emerged to describe the *building and maintenance of the connection between sensor data and the symbols used by a robot for abstract cognition*. As a matter of fact, anchoring is an important issue for any situated robot performing abstract reasoning based on physically grounded symbols. Amongst others, anchoring plays an important role to communicate or relate to either other robots or humans.

There are tasks, such as object manipulation or functional imitation, where anchoring requires explicitly recognizing objects and localizing them in the three-dimensional space. Fortunately, such an *object recognition* task is not necessarily required to achieve anchoring. In applications such as human/object tracking, face and object identification, or grounded robot-human communication, *object identification* is enough. Informally, to *recognize* an object often requires from the robot to match its percepts with a known model of the object[2]. This task has been studied for a few decades now and is known to be difficult in unknown environments. On the contrary, *identifying* the sole presence of an object is simpler since its goal is to *classify* or to *name* an object[3]. As a matter of fact, there exist many easy to use and reliable descriptions for characterizing the presence of an object. To identify the presence of a fire in a room, one does not have necessarily to visually recognize it. Smelling smoke, hearing cracks, feeling heat, seeing dancing shapes on a wall are different ways of identifying the presence of a fire. For an autonomous robot, the ability to identify objects is a first step towards more complex tasks and may be built by regularly checking for the object. Identifying objects is therefore a simple form of anchoring symbols (such as *fire*) to its percepts.

In this paper, we are concerned with a practical task, where a PIONEER 2DX mobile robot has to rely on its limited visual sensors to anchor symbols such as *human being*, *mobile robot* or *fire extinguisher* (etc.) that it encounters while navigating in our laboratory. Anchoring is then used to support human/robot or robot/robot communication. For instance, an interaction may be engaged if a *human being* is identified, or a rescue operation may be initialized if a non-responding PIONEER 2DX is identified. Identifying a *fire extinguisher* may allow the robot to respond to a query formulated by a human. To design an

autonomous robot, living in a changing environment such as our laboratory, with the identification ability described above is a difficult task to program. As such it is a good candidate for a Machine Learning approach, which may be easily recasted as a classical *concept learning task*. To teach the robot to anchor symbols using Machine Learning has proven successful[4]. To use machine learning techniques, the designer has to both define learning examples and a representation language based on the robot percepts to describe them.

It is clear that a great part of the success of the learning task per se depends on the representation chosen[5]. Having an AI designer providing the robots with an adequate representation has a major drawback: it is a fixed, ad-hoc representation. Any change of setting (a museum instead of an AI lab) may require a new perceptual description. In order to overcome this drawback, our main objective is to endow an autonomous robot with the ability to dynamically abstract from its percepts different representations, well suited to learn different concepts. The intuitive idea is to have the robot explore the space of possible examples descriptions (with various colors, resolution, representation formalisms, etc.) so as to discover for each concept a well-fitted representation. The underlying intuition being that for anchoring the symbol *human being* a robot does not need the same visual *stimuli* that might be necessary to identify a *power-plug* on a wall.

Section 2 presents a concrete setting in which this problem occurs and pinpoints why adapting one's representation may be useful to increase learning accuracy. Section 4 explains our approach based on abstraction operators applied to visual information provided by the robot. Finally in section 4 and 6, a set of real world experiments describes the interest of such an approach and outlines the difference between three representations, each one fitted to a different concept (the presence of a human, a fire extinguisher or a box).

2 Problem settings

2.1 The MICROBES Project

The practical task we are concerned with takes place in a wider project called MICROBES[6], which is a collective robotics experiment started in 1999 and involving more than 10 people. This project aims at studying the long-term adaptation of a micro-society of autonomous mobile robots in an environment populated by a human collectivity: the LIP6 laboratory in Paris. The robots, ten PIONEER 2DX, have to survive in this environment as well as cohabit harmoniously with its inhabitants.

From an individual point of view, they need to recharge themselves autonomously, build the map of their environment in order to memorize its characteristics and localize themselves, avoid the mobile obstacles (human beings, other robots) and the potentially dangerous places (stairs, lifts).

From a collective point of view, they have to solve the spatial conflicts (access to the charging stations, coordination in navigation), cooperate by sharing information about the environment (open or closed rooms, etc.) and abide by some individualized constraints in their interactions with human beings (e.g. learning individual schedules and respect for privacy).

The colony of robots does not have, then, a functional goal, apart from being able to survive in an eco-system in which it must implement a robust and adaptive social structure. Thus, by studying robots that are physically as well as socially situated, MICROBES works towards two main goals : design a sufficiently autonomous and versatile robotics basis that can be used in different applications (distributed surveillance of buildings, guidance of visitors, etc.) and study, in collaboration with sociologists, the conditions required for immersing autonomous mobile robots in a larger public.

2.2 *Anchoring and the building of a perceptual system*

Inside the MICROBES project, we are concerned with providing the robot with the ability to perform robot-human communication about objects in the world. However, from the robot's point of view, using a shared lexicon of human symbols requires some prerequisites such as grounding these symbols in order to make sense in the world[7].

In this paper, we aim at providing each PIONEER 2DX autonomous mobile robot with the ability to identify (*i.e.* correctly classify) objects or living beings encountered in its environment thanks to mechanisms inspired from perceptual learning. As stated in the introduction, there is a strong difference between object *recognition* and *identification* as stated in [3] :

- **object recognition** consists in finding a familiarity with an object for which there already exists a (*usually 3D*) model known by the system.
- **object identification** consists in classifying or naming an object. *i.e.* it requires neither a model of the object nor complex scene reconstruction algorithms.

This identification ability will serve to build a lexicon of grounded human symbols in order to provide a basis for human-robot interaction-based behaviors (*e.g. dialogue* using the lexicon, request to *track* or *follow* an anchored object, etc.). This paper focus on the anchoring process while the use of a lexicon for

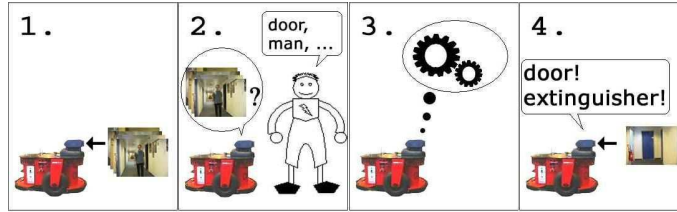


Fig. 1. The four steps toward lexicon anchoring. As a first step, the robot takes snapshots of its environment which are *labeled* by a supervisor. The robot tries to associate (learn) the provided label(s) with its percept, and, after a number of such steps take place, it shall be able to autonomously label a new environment.

such tasks will not be described here. It is important to understand that the anchoring process described here is independent of any behavior.

In practical, each robot navigates in the environment during the day and takes snapshots of its field of vision with its video camera according to three possible behaviors :

- **wander behavior** : the robot explores its environment and takes snapshots from time to time. This behavior is useful to get a set of images that is representative of the environment.
- **attention behavior** : the robot takes a snapshot upon a request. This enables a supervisor to show specific scenes.
- **active learner behavior**: the robot explores its environment and takes snapshots that are supposed to be interesting according to what it already knows. In machine learning, such *active learning* techniques can greatly improve the accuracy of new classifiers by selecting examples based on the performance of previously learnt classifiers.

At the end of each day, the robot may report to a supervisor and "ask" her/him what objects (whose symbols may or may not belong to a pre-defined lexicon) are to be identified on a subset of taken pictures (without the supervisor pointing at them). It then performs a learning task in order to create or update the connection between sensory data and symbols which is referred to as the anchoring process. From a machine learning point of view, the learning tasks produces classifiers that should then be used to identify symbols from the sensory data. Figure 1 describes this process. The learning task associated to the anchoring is therefore characterized by a set of image descriptions and attached labels. It corresponds to a multi-class concept learning task.

A key aspect of the problem lies in the definition of the learning examples (*i.e.* the set of descriptions extracted from the images) used by the robot during the anchoring process. In effect, a first step in any anchoring process is to identify (relevant) information out of raw sensory data in order to reduce the complexity of the learning task.



Fig. 2. Examples of the robot's visual experience.

The PIONEER2DX mobile robot provides images thanks to its LCD video camera while navigating in the corridors. The images are 160×120 wide, with a 24 bits color information per pixel. Humans, robots, doors, extinguishers, ashtrays and other possible targets can be seen among the images as shown in Figure 2. All these possible targets, as they appear in the images, are of different shape, size, orientation and sometimes they are partially occluded. Finally, each image is labeled with the names of the occurring targets.

3 Related works

In this section, we will briefly review two research domains that are more or less related to our problem setting. We will try to highlight both the specificity of our task and the common concerns between problem settings. Firstly, we will state the differences with works studying the *emergence of language* in a society of robots. Then, we will study common concerns between *content-based image retrieval* and our identification task.

3.1 Emergent adaptative lexicon and language

Lexicon anchoring is mainly concerned with studying the evolution of a language in a society of agents through the emergence of a shared grounded lexicon. In order to build a shared lexicon, a group of agents may require a combination of individual adaptation, cultural evolution, and auto-organisation[8].

These works do not focus much on the problem of extracting visual percepts. The world is perceived through few "channels" (such as *color*, *localization*, *height*, *width*) and *discrimination trees* are built incrementally to disambiguate words[9]. In fact, the problem of perceptions is simplified so that it is possible to study the evolution of language on a large scale (*i.e.* grounding meaning in a society of agents).

While these works achieved very interesting results and deals with the ground-

ing of a lexicon, we are not concerned with the same issues. As a matter of fact, we consider the anchoring of a *given* lexicon, i.e. how to extract relevant information from complex images, instead of the *emergence* of such a lexicon, i.e. lexicon adaptation, evolution of grammar or syntax, etc.

3.2 Content-based image retrieval

Our problem setting shares much more in common with that of *content-based image retrieval*[10] (or CBIR). However the goal in CBIR is (roughly) to compute a similarity measure between two images, the question as to how the information is extracted remains central in both cases. As a matter of fact, we can learn much by studying the popular approaches used in CBIR to describe an image. There are three main approaches based on:

- **Global color histogram description**[11] : matching image's histogram descriptions achieved surprisingly good retrieval results and is considered as a benchmark to evaluate other approaches. This approach is simple yet efficient.
- **Region-based similarity**[12,13] : the similarity measure is computed by matching regions grown according to various properties of the images (*e.g. color and texture properties*). However good results were achieved using this approach, there are known drawbacks such as the complexity of matching between images described as sets of regions and the unreliability of region-growing algorithms.
- **Configural recognition**[14] : this approach provides an efficient way to compare images using spatial properties between regions while limiting the matching complexity. Only a *fixed* number of regions according to a *given* configuration are taken into account. Since the *template* is given by the supervisor², this approach is fitted for retrieval of images with constant overall organizations (*i.e. scenes (e.g. mountain, sea, etc.) vs. objects*).

The cited approaches can help us defining an image description mechanism but we should also take into account that there are strong differences between anchoring and CBIR. These fundamental differences are that :

- (1) *Retrieval is not identification* : CBIR uses a similarity measure that do not explicitly classify the example. Moreover, learnt classifiers (set of rules, decision trees, trained neural networks, etc.) are faster to apply than computing any similarity measure (*i.e. one-pass test vs. complex matching phase*). Such classifiers enable nearly costless image classification and can easily be implemented in a real-time operating mobile robot.

² The values for each component within the fixed template can be also learnt[15].

- (2) *CBIR is not a long-term behavior* : the robot is supposed to navigate in the environment and constantly update its anchors. Since the world is dynamic and subject to *concept drifts*, the robot requires to be able to learn and adapt its anchors through time (*e.g.* if a new example of the "chair" symbol may appear someday).
- (3) *The images are not collected thanks to a situated behavior* : the data collected by the robot are specific to its location. Due to the properties of such images, we are concerned with checking if there is a specific property hidden in the image that would help to identify an object. As a matter of fact, the environment of the robot provides very similar images where global variations are not bounded to a given object. On the other hand, CBIR is about retrieving globally similar images among a set of very different images.

4 Changing the Representation of Images

4.1 Initial Perceptual Representation

We define the role of the robot's perceptual system as to extract *abstract percepts* out of *low-level percepts*, such as a set of pixels, from the video camera or sonar values. These abstract percepts provide a representation of the perceived world on which further computation will be based. They can be anything from sets of clustered colored regions to a matrix resulting from a Hough transform. The choice of a representation is motivated by finding a good trade-off that reduces the size of the search space and enhances the expressiveness of the abstract percepts.

As mentioned in section 2, the problem we consider is that of automatically finding a representation of a set of labeled images that is well adapted to the learning of concepts. Let us underline that our goal is not to achieve the best performance on the particular learning task mentioned in the previous section. To obtain the best performance would require that experts in the field build an ad-hoc representation for each concept to learn. On the contrary, we are interested in having a robot find by itself the good representation, so that, if the context changes or the concept to learn is different, it has the ability to discover by himself the good level of representation. We therefore consider the representation provided by the sensors as an *initial* representation.

From the robot's point of view, each pixel from the camera is converted into a *low-level percept*. In the initial image representation, where each pixel is described by its position (x,y), its *hue* (the tint of a color as measured by the wavelength of light), its *saturation* (term used to characterize color purity or

brilliance) and its *value* (the relative lightness and darkness of a color, which is also referred to as "tone"). The initial description of an image is therefore a set of 19200 (160 x120 pixels) 5-tuple (x,y,h,s,v). Each image is labeled by symbols following the process described in Section 2 (see also fig. 2). The *positive* examples of a given concept (e.g. "presence of a fire extinguisher") to learn correspond to all images labeled positively for *this* concept. The *negative* examples are the images not labeled for *this* concept. As a matter of fact, a negative example for a given concept can be a positive example for another concept. The number of positive examples for each concept may vary greatly depending on the environment, the exploration of the robot, etc.

The initial representation of images, consisting of hundreds of thousands of pixels, is clearly a too low-level representation to be used by Machine Learning algorithms. We shall now analyze different representations that have been considered in the field of Computer Vision from the Machine Learning point of view. These different representations will provide some directions for investigating automatic changes of representation to improve the learning accuracy.

4.2 Representation Languages in Machine Learning

In the traditional setting of Machine Learning, each object is represented by a *feature vector* x , to which is associated a label $f(x)$. The supervised learning task consists in finding a classifier h which minimizes the misclassification probability $\Pr[f(x) \neq h(x)]$ on a newly observed example $(x, f(x))$.

Within the *multiple instance* setting[16], objects are represented by *bags of feature vectors*. Feature vectors are also called *instances*, as in the traditional setting features may be numeric as well as symbolic features. Again, the associated learning task consists in finding a classifier h_{multi} such that most bags are correctly classified. In this setting, multiple-instance classifiers are of the form $h_{multi}(b) = h(x_1) \vee \dots \vee h(x_r)$ where $b = \{x_1 \dots x_r\}$ is a bag containing r instances. Thus, an object represented by a bag b will be classified positively by h_{multi} iff at least one of its instances fires h . Multiple-instance learning has been successfully applied to various domains including the prediction the chemical activity of molecules[16], and the classification of natural scenes[15].

Within a *relational setting* the objects are represented by a set of components objects, their features, and relations between components. In particular, in Inductive Logic Programming[17] Prolog facts are used to describe objects and Background Knowledge B encodes deductive rules.

To summarize, in Machine Learning the languages used to represent examples fall into three broad categories:

- **Feature-Vector:** the most widely used and for which efficient algorithms have been devised.
- **Relational description:** the most expressive representation but whose inherent complexity[18] prevents from efficient learning.
- **Multiple-instance:** an in-between representation, more expressive than feature-vector but for which efficient algorithms do exist.

4.3 Dimensions of abstraction

In the perspective of automatically exploring the set of possible representations of an image, we propose to identify particular operators and to experiment with them. There are countless operators that could be applied to an image hoping for more accurate learning. Operators changing the *contrast*, the *resolution*, the *definition* are all possible candidates.

To improve the learning of concepts, we are interested in transformation that are *abstractions* in the sense that they decrease the quantity of information contained in the image[5]. Abstraction is considered as a specific *change of representation* that is an *homomorphism* from one representation to another (here : from an image to its description). Starting from the initial *low-level percepts* (*i.e.* the pixels of the image), the elements obtained after applying the *abstraction operators* will be referred to as *abstract percepts*, since they will be used as representative percepts for further processing.

The two main dimensions of abstraction that we shall study are **granularity** and **structure**. Granularity corresponds to the resolution of the image. Structure corresponds to the basic element of the image as the smallest individually accessible portion of the image to consider, be it a pixel or a complex region. Figure 4 depicts the space of representation changes associated to these two dimensions and their corresponding abstraction operators that we define as :

- The **associate operator** (for granularity) : it consists in replacing a set of pixels with a unique (mega)pixel that has for its (h,s,v) values the average of the pixels that were associated. This operator is a built-in operator for the robot as it corresponds to a particular *sub-sampling*. The resulting *abstract percepts* will be referred to as *r-percepts*³.
- The **aggregate operator** (for structure) : it consists in grouping a set of pixels or regions to form a pattern. This operation is also referred to as "term construction" in the literature[19]. The pattern does not replace the pixels or regions it is composed of, and therefore the resolution or granularity of the image is not changed. What changes is the structure of the image. The aggregate operator may be either data-driven (*e.g.* growing patterns)

³ r-percepts, as in *resolution percept*.

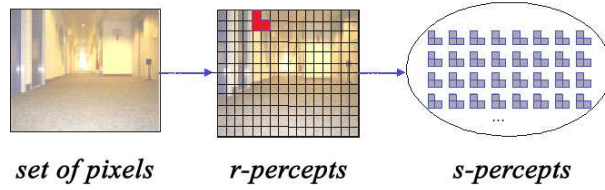


Fig. 3. An example where specific instances of the operators *associate* and *aggregate* are sequentially applied to an image.

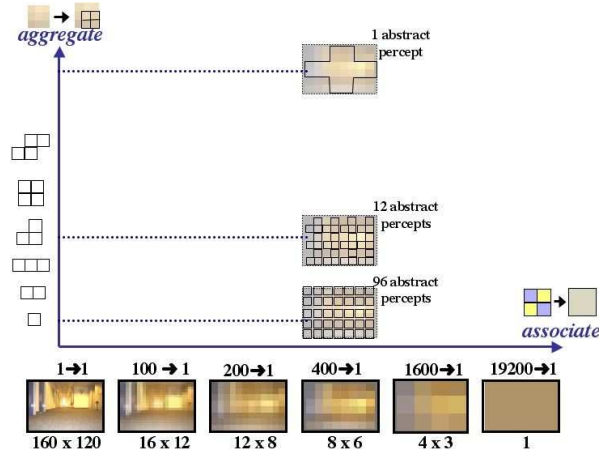


Fig. 4. The space of image representation obtained by applying the associate operator (changing the *resolution*) and the aggregate operator (changing the *structure*).

or model based (*e.g.* applying a predefined mask). For reasons of efficiency required by the use of a robot we have considered an aggregate operator that is applied to contiguous pixels forming a particular shape (we do not consider region-growing algorithms because of their versatility when using fixed thresholds). The resulting *abstract percepts* will be referred to as *s-percepts*⁴.

Figure 3 illustrates how one can use these operators by showing a practical example where specific instances of the associate and aggregate operators are sequentially applied to extract a new description from an image.

5 Automatically changing the representation for learning

In the previous section two abstraction operators to change the representation of images were presented. The parameter of the associate operators we have considered is the number of pixels that are associated to form a (mega)pixel. The parameter of the aggregate operator is the pattern or region structure.

⁴ s-percept, as in *structural percept*.

With respect to the learning task described in Section 2, a key issue is to analyze the impact of representation changes on learning. The main question is related to the choice of one operator and its parameters. In Machine Learning, the abundant literature on feature selection shows that approaches fall in two broad categories: the **wrapper** and the **filter** approach [20]. Intuitively, the **wrapper** approach uses the performance of the learning algorithm as a heuristic to guide the abstraction. In the following, we present how the **wrapper** approach can be used to choose the most fitted abstraction. As it is an approach that attempt to learn from the learning process itself it is also referred to as a *meta-learning* approach.

We have developed the PLIC system, which is both an image description toolkit, a data reformulation tool, and a wrapper. PLIC interacts with RIPPMI, a *multiple instance rule learner* that generates classifiers as decision rules (see [21] for a full description of RIPPMI). For example, a typical classifier would be (using a s-percept such as the one seen in figure 3) :

- HYPOTHESIS: HUMAN.
- TRUE :- P3VALUE<=9, P2SATURATION>=27.
- TRUE :- P2HUE<=203, P1SATURATION<=3, P3VALUE<=165.
- TRUE :- P3HUE <=198, P1X>=6, P1Y>=2.
- DEFAULT FALSE.

Where P1, P2, P3 are the corresponding embedded r-percepts. RIPPMI cross-validates learnt classifiers in order to evaluate the average error rate on unknown data. This is generally a reliable estimation of the classifier’s accuracy on future data.

With the help of RIPPMI, PLIC applies the operators as follow :

- (1) **association operator** : The horizontal dimension in figure 4 is difficult to explore since object identification is independent of scaling. Since there is no ”better” resolution and that every resolutions should be useful, PLIC describes each image by using the associate operator for several image resolutions (namely 1x1, 4x3, 8x6, 16x12, 32x24⁵). The idea behind this *multi-granularities approach* is to learn classifiers that are invariant to resolutions and object size variations.
- (2) **aggregation operator** : PLIC uses its wrapper component in order to explore the vertical dimension in figure 4. Exploring the vertical dimension is used to select between different structural patterns to apply with the aggregate operator. The wrapper-based component explores different s-percepts iteratively as synthesized in figure 5. An initial s-percept is chosen (at first, it embeds only one r-percept), and the image is reformulated in a multiple-instance representation using this structure; then, the

⁵ We were not able to go further due to memory limitations.

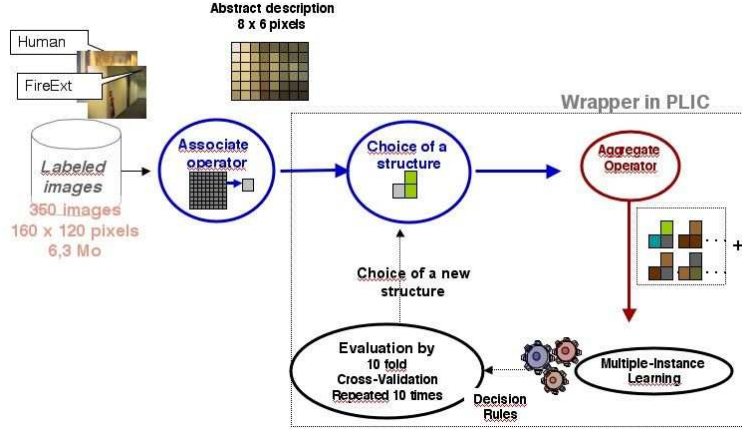


Fig. 5. The PLIC system Wrapper component.

concepts are learnt using this representation. Based on the results with cross-validation of the learning algorithm, a new structure is devised by adding a contiguous r-percept. The heuristic for creating a new structure is based on the fact that for the current s-percept, all the embedded r-percepts are used in at least one decision rule of the rule set with the accuracy being better than at the previous level. For example, the ruleset we saw before would be extended (all three embedded r-percepts are used).

PLIC uses RIPPMI to learn several *classifiers* for each object to be identified (*e.g.* one classifier for each s-percepts). Each classifier is learnt thanks to a fixed number of positive and negative examples during a **batch learning session**. However these classifiers are cross-validated, it is possible that the robot may encounter new occurrences of the object. For example, people may change clothes, objects may be moved or replaced, the environment can vary greatly during the day (*e.g.* daylight vs. artificial light), etc.

Fortunately, all these classifiers can be combined in order to evaluate which ones have to be replaced. PLIC addresses **anchoring in the long term** by increasing or decreasing each classifier's *weight* depending on the accuracy of its prediction when new images are presented. Many on-line learning algorithms can be used such as the weighted majority algorithm[22] or even a simple perceptron. As a consequence, we can easily replace outdated classifiers by the newly learnt classifiers whenever there is a batch learning session. For a given concept, such a session can be launch once n new images have been labelled with this concept (choosing n is free but may take into account memory limitation since this is a batch learning task where all positive and negative examples are handled at the same time as opposed to on-line learning).

6 Experiments

6.1 Experimental Setup

To evaluate the interest of abstracting visual percepts from a Machine Learning point of view, a number of different experiments have been carried out. The experiments presented are based on the images acquired by a PIONEER2DX mobile robot in the corridor of the *LIP6* laboratory (Paris, France). The objects as they appear in the images are different in shape, size, orientation and are sometimes partially occluded. The lexicon contains three symbols to anchor:

- (1) "human": a single person with different kind of clothes.
- (2) "fire extinguisher": they can be found in the corridor of our lab.
- (3) "box": various boxes that stand alone or piled up.

As explained in Section 2, a supervisor "names" the occurring targets. Given the symbol to anchor, we have decided that every batch learning session would be based on the first 25 positively labeled examples and 25 randomly selected negative examples (no bias). The size of the corresponding descriptions depends on the operators. For example, a learning set may vary from approx. 2Kb (with a global histogram description for each image) to 3Mb (with an associate operator set to 32x24 and an aggregation operator with s-percepts that embed 4 r-percepts). Among the positive examples, about 50% are labeled with one objet, 15% with two objects and 5% with three objects.

Three independent sets of experiments are presented. The first one illustrates the impact of the operator associate used to build a multi-granularities descriptions. The second studies the impact of the aggregate operator based on an arbitrarily selected granularity. The multiple instances rule learner RIPPERMI was used on the descriptions obtained from these images with a ten-fold cross validation⁶. Moreover, each experiment is repeated 10 times in order to get a good approximation of the results. In machine learning, such a validation is known to compute a good approximation of what will be the real accuracy of the classifiers (*i.e.* the object identification accuracy). RIPPERMI returns a set of rules (*i.e.* a classifier) that covers the positive examples. Finally, we describe the use of weights to evaluate classifiers obtained at the previous steps and show the benefits of updating the anchor of an object through time.

⁶ Cross-validation is a widely used data-oriented evaluation of the learning generalization error. The data set is divided into a learning and a training set.

	"human"	"extinguisher"	"box"
<i>global histogram (baseline)</i>	<i>61.83%</i>	<i>64.72%</i>	<i>77.78%</i>
4x3	50%	63.89%	69.17%
8x6	50.28%	66.67%	86.11%
16x12	63.61%	58.61%	82.5%
32x24	72.5%	70.28%	56.67%
multi-granularities	<i>72.8%</i>	<i>75.28%</i>	<i>75.8%</i>

Table 1

object detection accuracy and granularity.

6.2 Evaluating automatic changes of granularity

To begin with, we performed a simple learning task using RIPPER[23], a well-known supervised learning algorithm, with a learning set consisting of the popular⁷ global histogram descriptions of the images. This will serve as a baseline to evaluate the impact of choosing a specific granularity.

Table 1 shows the object identification accuracy for the three concepts⁸. According to the results, it is not clear which resolution is better. The experiment with the global histograms sometimes yields better results than experiment with finer grain. Moreover, the multi-granularities approach seems to yield only slightly better results than other approach and is even worse for the easy-to-learn "box" concept. Nevertheless, the multi-granularities approach produces classifiers that are resolution independent: each classifier is learnt on a dataset where each image is described in four different ways (*i.e.* 4x3, 8x6, 16x12 and 32x24 representations) that generates four distinct examples.

Clearly, object identification depends on the object and its accuracy is subject to change through time and experience. While the multi-granularity approach do not always yield the best results, there are good chances that its classifiers will prove more robust in time than other classifiers.

6.3 Experiments on automatic changes of structure

PLIC's wrapper tool was used with the heuristic described in section 5 in order to generate up to a maximum of 4 r-percepts per s-percept. The possible structural configurations are shown in figure 6. Each structural configuration is

⁷ At least in CBIR.

⁸ Learning duration is less than 10 sec. standard deviation is about 1%.

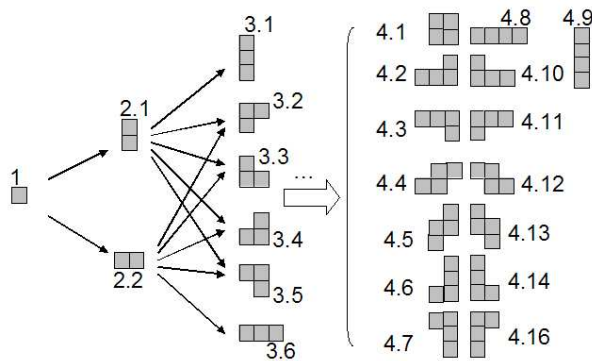


Fig. 6. Four levels of structural configurations (i.e. s-percepts) generated by PLIC

	"human"	"extinguisher"	"box"
<i>global histogram (baseline)</i>	61.83%	64.72%	77.78%
level 1	72.5%	70.28%	56.67%
level 2	73.5% (2.2)	73.33% (2.2)	60% (2.2)
level 3	76.67% (3.5)	73.33% (3.4)	76.7% (3.4)
level 4	80.8% (4.10)	80.8% (4.3)	85.33% (4.16)

Table 2

object detection accuracy: best results for each structure level.

applied from every single r-percept to generate the learning sets. The 32×24 resolution was chosen in order to show the potential of structural reformulation. Table 2 shows the best results achieved for each structural level of complexity.

Results from the experiments show that for all the objects, the highest accuracy is achieved by one of the most complex structural configurations, which is not surprising. However the structural configurations are still quite simple, the identification accuracy for each object rose between 8 points (human detection) and 29 (!) points (box detection).

Eventhough the impact of modifying the aggregation operator depends on the concept to learn (same as the association operator), structural reformulation is clearly an efficient way to improve classifiers for anchoring. The classifier shown in section 5 was learnt from a reformulated dataset using the "3.3" s-percept. This classifier demonstrates that relations between the embedded r-percepts are taken into account.



Fig. 7. Three snapshots taken during a tracking behavior (with identification).

6.4 On-line learning and updating anchors

We saw previously that PLIC and RIPPERMI require a fixed number of examples during a *batch learning session*. As a consequence, these *batch learning algorithms* build efficient classifiers as long as examples are representative of the world. Given the world is dynamic and unstable, we have to update the anchors from time to time. An interesting approach is to combine an on-line learning algorithm such as the well known *weighted majority algorithm*[22] with our batch learning algorithms. Such an on-line learner would:

- provide a global detection prediction by aggregating the weighted classifiers predictions.
- improve the global performance of a set of classifiers in the long term by evaluating them (classifiers with a low accuracy are replaced by new ones).

We experimented this algorithm during several batch learning sessions and it proved to be efficient thanks to the following characteristics:

- (1) it *naturally improves classifiers*: Each learning session is based on a specific set of data. If the robot's environment is (more or less) stable, it is possible to grasp new object's properties or to take better snapshots. This sometimes results in learning better classifiers that can slightly increase the global identification accuracy for an object. In our experiments, we empirically evaluated this as less than a 5% growth in object identification accuracies for different kind of redundant objects.
- (2) it performs *long-term adaptation to concept drifts*: We experimented on the tracking by identification of a human being dressed in grey and black. The robot built its classifiers during two learning sessions. Then, the robot was made unable to track the target because the human dressed in blue and white. After two other learning sessions, new classifiers were built and the robot could track the human target again. What is important here is that some of the old classifiers still remained. These few classifiers relied on *skin* and *hair* colors, which are constant human features.

Figure 7 shows three snapshots taken during tracking human (and other objects). Different classifiers were used depending on the image. On the first

image, classifiers identified a human based on *t-shirt*, hair and skin colors using different structures. A box is also identified. On the second image, human detection relies simply on the color of the skin. Finally, the third image shows an example of wrong detection on the right part of the picture due to an unknown environment (bureau *vs.* corridor). Nevertheless, the human is also detected thanks to skin-based classifiers and a "t"-like structure classifier that covers the face (skin and hair).

7 Conclusion

In this paper we have addressed the problem of using automatic abstraction of visual percepts by an autonomous mobile robot to improve its ability to learn *anchors*[1]. This work finds its application in a real-world environment within the MICROBES multi-robots project[6], where anchors provides a basis for communication between the PIONEER 2DX robot and its human interlocutor. In the approach we proposed the robot starts with the initial low-level representation of the images it perceives with its LCD video camera, and iteratively changes their representation so as to improve the learning accuracy. Between the low-level pixel representation and a global histogram representation there is an immense space of possible representations. To explore part of this abstract space of representation we have identified two operators. A first one changes the resolution and loose information by averaging the color of squares of pixels. A second one that groups pixels without changing the resolution.

To guide the exploration of the space of possible abstractions, we have developed the PLIC system which uses the learning results in order to select the abstract operators to be applied. From a Machine Learning point of view, this architecture is based on a widely used approach in feature selection: the Wrapper-model. The set of experiments that have been conducted show that both operators do impact on the learning accuracy. It is interesting to notice that the best resolution and structure (sort of coordinates in the abstract space) found by the system depends of the concept. It is also clear that as the number of examples increases, different reformulations might perform better. Creating high-level abstract percepts does not only improve accuracy, it makes object identification faster for the robot. This is true as long as the abstraction process does not itself takes too much time. This is a known trade-off in the field of abstraction[24]. As a matter of fact, abstracting regions by using region growing algorithm was a candidate abstract operators but its computation is too costly for online identification.

This study shows that for learning anchors, an approach that periodically searches for the most accurate representation, given the examples at hand, is a promising direction. Moreover, it appears that for each anchor that needs

to be learnt, different abstractions might be more appropriate. These findings raise several questions with respect to the robot architecture. The search for a better representation should be triggered by a decrease of performance of the acquisition of new examples? How to compare the application of operators that change the resolution and operators that change the structure? A central question for any lifelong learning system, integrating abstraction abilities, is to decide whether to continue to *exploit* its current representation or *explore* new representations at the risk of losing resources if no better ones is found.

Acknowledgments

The MICROBES project is supported by three grants from the French Department of Higher Education and Research, by a grant from the LIP6 for the collaboration between research teams, and by a research contract between the LIP6 and France Telecom R&D.

References

- [1] S. Coradeschi, A. Saffiotti, Anchoring symbols to sensor data: preliminary report, in: Proceedings of AAI-2000, Austin, Texas, July 2000.
- [2] D. Marr, Vision, Freeman and Co., Oxford, 1982.
- [3] J. Stone, Computer vision: What is the object?, in: Prospects for AI, Proc. Artificial Intelligence and Simulation of Behaviour. Birmingham, England., IOS Press, Amsterdam. pages 199–208, 1993.
- [4] V. Klingspor, K. Morik, A. D. Rieger, Learning concepts from sensor data of a mobile robot, Machine Learning 23 (2-3) (1996) pp305–332.
- [5] L. Saitta, J.-D. Zucker, A model of abstraction in visual perception, Applied Artificial Intelligence 15 (8) (2001) pp761–776.
- [6] S. Picault, A. Drogoul, The microbes project, an experimental approach towards open collective robotics, in: Proc. of the 5th Int. Symposium on Distributed Autonomous Robotic Systems, Springer-Verlag Tokyo Inc., 2000.
- [7] S. Harnad, The symbol grounding problem, Physica D (42) (1990) pp335–346.
- [8] L. Steels, Emergent adaptive lexicons, in: In P. Maes, editor, Proceedings of the Simulation of Adaptive Behavior Conference. MIT Press., 1996.
- [9] L. Steels, Perceptually grounded meaning creation, in: Proceedings of the First International Conference on Multi-Agent Systems, 1996, pp. pp338–344.

- [10] J. Eakins, M. Graham, Content-based image retrieval, in: Report to JISC Technology Applications Programme, Institute for Image Data Research, University of Northumbria at Newcastle., 1999.
- [11] M. Stricker, M. Swain, The capacity and the sensitivity of color histogram indexing, in: Technical Report 94-05, Communications Technology Lab, ETH-Zentrum, 1994.
- [12] J. Z. Wang, Integrated Region-based Image Retrieval, The Kluwer International Series on Information Retrieval, 11, Oxford, 2001.
- [13] S. Belongie, C. Carson, H. Greenspan, J. Malik, Color- and texture-based image segmentation using em and its application to content-based image retrieval, in: Proc. Int. Conf. Comp. Vision 1998., 1998.
- [14] P. Lipson, E. Grimson, P. Sinha, Configuration based scene classification and image indexing, in: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), IEEE., 1997.
- [15] O. Maron, A. Ratan, Multiple-instance learning for natural scene classification, in: Proc. 15th ICML, 1998, pp. pp341–349.
- [16] T. G. Dietterich, R. H. Lathrop, T. Lozano-Pérez, Solving the multiple-instance problem with axis-parallel rectangles, *Artificial Intelligence* 89 (1-2).
- [17] S. Muggleton, Inductive logic programming, *New Generation Computing* 8 (4) (1991) pp295–318.
- [18] A. Giordana, L. Saitta, Phase transitions in relational learning, *Machine Learning Journal* 41 (2) (2000) pp217–241.
- [19] A. Giordana, L. Saitta, Abstraction: a general framework for learning, in: Working notes of the AAAI Workshop on Automated Generation of Approximations and Abstraction, Boston, MA, 1990, pp. pp245–256.
- [20] R. Kohavi, G. John, The wrapper approach, in: Feature Selection for Knowledge Discovery and Data Mining, H. Liu and H. Motoda (eds.), Kluwer Academic Publishers, pp33-50., 1998.
- [21] Y. Chevaleyre, N. Bredeche, J.-D. Zucker, Learning rules from multiple instance data : Issues and algorithms, in: Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU02), Annecy, France., 2002.
- [22] N. Littlestone, M. Warmuth, The weighted majority algorithm, in: IEEE Symposium on Foundations of Computer Science, 1989, pp. pp256–261.
- [23] W. W. Cohen, Fast effective rule induction, in: Proc. 12th International Conference on Machine Learning, Morgan Kaufmann, 1995.
- [24] F. Giunchiglia, Using abstrips abstractions : Where do we stand ?, *Artificial Intelligence Review* 13 (1996) pp201–213.