Journal of Universal Computer Science, vol. 10, no. 8 (2004), 1014-1033
submitted: 22/3/04, accepted: 28/6/04, appeared: 28/8/04 © *J.UCS*

Galois Lattice Theory for Probabilistic Visual Landmarks

Emmanuel Zenou
(SUPAERO & LAAS-CNRS, Toulouse, France
zenou@supaero.fr)

Manuel Samuelides
(SUPAERO, Toulouse, France
samuelid@supaero.fr)

Abstract: This paper presents an original application of the Galois lattice theory, the visual landmark selection for topological localization of an autonomous mobile robot, equipped with a color camera. First, visual landmarks have to be selected in order to characterize a structural environment. Second, such landmarks have to be detected and updated for localization. These landmarks are combinations of attributes, and the selection process is done through a Galois lattice. This paper exposes the landmark selection process and focuses on probabilistic landmarks, which give the robot thorough information on how to locate itself. As a result, landmarks are no longer binary, but probabilistic. The full process of using such landmarks is described in this paper and validated through a robotics experiment.
Key Words: computer vision, visual landmarks, localization, orientation, autonomous mobile robotics, Galois lattices, concept lattices, formal concept analysis.
Category: H.3.7, H.5.4

1 Introduction

Finding its place in an environment is a difficult challenge today for an autonomous mobile robot. This robot needs to know how to characterize and to recognize a place by itself in order to be considered fully autonomous in terms of orientation and navigation.

The robot in our experiment uses visual landmarks (section 2) to characterize each site in the structural environment. This localization approach means that the robot recognizes its topological but not metric position. Thus it is more qualitative than quantitative. Metric localization is efficient in a local context, provided objects are not moving. But as soon as the environment gets larger, or objects start moving (as in a typical human environment), metric localization becomes very costly and less stable than the topological approach. Moreover, the use of visual landmarks is justified through advantages the sensor (a standard "webcam" in our application) presents: small, cheap, robust. By nature, images are also -potentially too- replete with information.

The formalism used to select landmarks is *Galois* -or *concept- lattice* (section 3), and a general approach has already been described in a previous paper [Zenou and Samuelides, 2003]. We have developed a general Galois lattice-based landmark selection algorithm. Here we describe this technique much more thoroughly than previously (section 4). Moreover, we use the *heritage lattice* (see section 6) to avoid attribute redundancy.

The robot thus locates itself if it finds a landmark specific to its site in the current image. However, this landmark-based localization process gives a weak *response rate*, *i.e.* a weak number of images that the robot knows how to locate compared with the total number of images it captures.

Our original approach developed in this paper is the introduction of *"probabilistic landmark"* (section 5) to improve the response rate. Most of the time, the probabilistic side of localization concentrates on the metric localization (see [Thrun et al., 2000, Thrun, 2002]), where landmarks are mainly interest points [Harris and Stephens, 1988]. Here, landmarks give a probability associated with each site in the structured environment.

In a practical way, if the robot does not find any classical landmarks, it will try to find such probabilistic landmarks. However, probabilistic landmarks must be very robust. Here we underline one specificity of our application. Indeed, if the robot cannot locate itself by finding (classical or probabilistic) landmarks in the images it captures, it will be able to move and catch other images until a robust landmark is detected. Thus it is an active vision process.

The landmark selection procedure (similar to a landmark-based classification process) and the localization process are described respectively in sections 6 and 7. Experimentations and results are exposed in section 8, before our conclusion in section 9.

2 Visual Landmarks

In [Zenou and Samuelides, 2003], a wide discussion is initiated to answer the following questions: *what is a landmark ? How to find it ? And how to select it ?* A new classification systems is introduced: *fully pre-defined landmarks*, the robot "just" has to recognize [Knapek et al., 2000, Mata et al., 2001]; *partially pre-defined landmarks*, for which a general structure is pre-defined, as in [Hayet et al., 2002] where authors use quadrangular surfaces (posters); and *non pre-defined landmarks*, where no hypothesis is done about landmarks (such landmarks are usually bio-inspired [Itti et al., 1998, Arleo et al., 2001]). Our approach belongs to this last category.

During the learning phase, all pictures are shot from the robot camera in the different places of the environment. Thus, a set of images is attached to each place of the environment. From these pictures, primitives (segments from polynomial contour extraction, color from the HSV space, objects quantified with morphological operators. . .) are extracted to find features, which are supposed to be more robust than primitives. The algorithms used to extract primitives from images are quite classical. For instance, to obtain segments, the contours are extracted with a Canny-Deriche algorithm and approximated with polynomial figures, from which segments are extracted (see Figure 1 as an example).

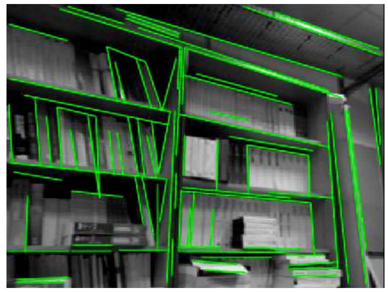


Figure 1: Segments issued from a shelf.

Color information is coded into the HSV space, that is much more robust against luminosity alteration. The three primary colors (red, green, blue) and secondary colors (yellow, magenta, cyan) pixels are extracted and objects (*i.e.* homogeneous local area with the same color) extracted.

To quantify such an information, morphological operators are used in the six binary images. Indeed, given for instance any image (Figure 2(a)) and the associated real "red" image ($H < 0.07$, $S > 0.7$, $V > 0.1$ in the normalized HSV space, see Figure 2(b)), after being "cleaned" with the open and close operators, there is a (minimum) 40×10 sized rectangle red object in the original image *if and only if* the result of an erosion is not an empty image (Figure 2(c)).

Features are issued from primitives. For instance, *"there is a yellow object"* or *"there is a large number of identical (orientation and size) segments"* (shelf) are typical features, more robust against rotation, translation and scaling than primitives. By this way, to each image is associated a set of features, and a *mapping* is filled up (Figure 3).

A cross means that the feature f (called "attribute" since now) is present in the image i (called "object" since now). The mapping is thus partitioned into classes. Our goal is to create a landmark-based classifier.

To mine information, an original mathematical formalism is used by the machine to find landmarks of its environment.

3 Galois Lattices

All information is described in terms of *concepts*, that associate sets of objects (images) with sets of attributes. This formalization matches very well with our

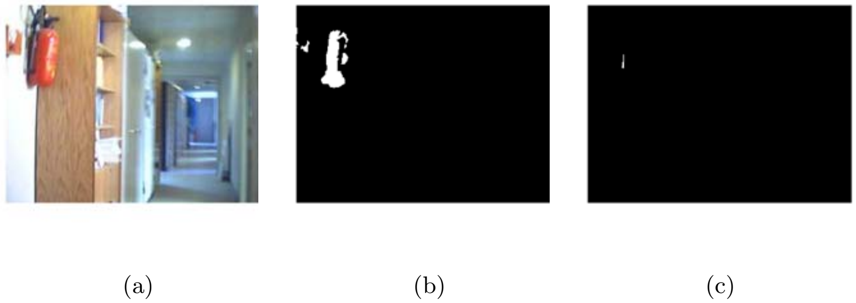


Figure 2: Colored Object Extraction: from the image (a), red colored pixels are extracted (b). Afterwards, morphological filters (*opening* and *closing*), using and small structuring element (SE) are done to clean the binary image, *i.e.* to remove small black pixels in the white space and small white pixels in the black space. Finally, An *erosion*, using a 50×20 SE is done on the filtered image and, as a resulting image (c) is not fully black, it shows the presence of a minimum 50×20 red object on the image.

application: subsets of images corresponding to particular places (inside a site) are described with particular feature combinations. For instance, a green plant is described as a green object with many small segments, inside a site. There is thus a very close relationship between landmarks, sites, images and concepts, which is not the case with classification trees for instance.

Moreover, updating a lattice and visual landmarks is very easy with an incremental algorithm. it is not necessary to restart the learning process when a new object arrives, in opposition with a neural network for instance. Moreover, concept lattices are fully deterministic, and do not depend on any stochastic initialization process.

All concepts are organized into a a lattice-type hierarchy, called Galois -or concept- lattices. Landmarks are extracted from these “hierarchized” concepts.

The full formalism about Galois lattices is described in the following books: [Barbut and Monjardet, 1970, Ganter and Wille, 1999]. Let us expose however the main definitions.

3.1 Galois Lattice formalism

Definition 1 *A* **lattice** *is a partially ordered set in which two any elements have a least upper bound (lub) and a greatest lower bound (glb). A complete lattice is a lattice where any set has a lub and a glb.*

		Feature 1	Feature 2	Feature 3	Feature 4		Feature f		Feature Nf
	Image 1.1	X	X						
	Image 1.2		X	X			X		
Place 1	Image 1.3								
			X						
	Image 1.N1								
	Image 2.1			X					
	Image 2.2			X	X	X		X	X
Place 2	Image 2.3					X			
	Image 2.N2								
...	...		X					X	
	Image p.1								X
	Image p.2		X					X	
Place p	Image p.3								
							X		
	Image p.Np								

Figure 3: Information structuring.

Definition 2 *A* **context** $\mathcal{K}$ *is a triple* $(\mathcal{O}, \mathcal{F}, \zeta)$ *where* $\mathcal{O}$ *is a set of* **objects***,* $\mathcal{F}$ *is a set of* **attributes** *and* ζ *is a* **mapping** *from* $\mathcal{O} \times \mathcal{F}$ *into* $\{0, 1\}$*.*

Definition 3 *Given a context* $\mathcal{K} = (\mathcal{O}, \mathcal{F}, \zeta)$ *let us define two mappings from* $\mathcal{P}(\mathcal{O})$ *into* $\mathcal{P}(\mathcal{F})$ *and from* $\mathcal{P}(\mathcal{F})$ *into* $\mathcal{P}(\mathcal{O})$ *using the same notation* $'$ *by the formula*

$$\forall \mathcal{A} \subset \mathcal{O}, \mathcal{A}' = \{f \in \mathcal{F} \mid \forall o \in \mathcal{A}, \zeta(o, f) = 1\} \tag{1}$$

$$\forall \mathcal{B} \subset \mathcal{F}, \mathcal{B}' = \{o \in \mathcal{O} \mid \forall f \in \mathcal{B}, \zeta(o, f) = 1\} \tag{2}$$

$\mathcal{A}'$ *is called the* **dual** *of* $\mathcal{A}$*, similarly* $\mathcal{B}'$ *is called the* **dual** *of* $\mathcal{B}$*.*

Definition 4 *Given a context* $\mathcal{K} = (\mathcal{O}, \mathcal{F}, \zeta)$*, the pair* $\mathcal{C} = (\mathcal{A}, \mathcal{B})$ *is called a* **concept** *of* $\mathcal{K}$ *if and only if* $\mathcal{A}' = \mathcal{B}$ *and* $\mathcal{B}' = \mathcal{A}$*.*

Definition 5 $\mathcal{A}$ *is called the* **extent** *of the concept* $\mathcal{C}$ *and* $\mathcal{B}$ *is called its* **intent***. This is denoted by* $\mathcal{A} = \texttt{extent}(\mathcal{C})$ *and* $\mathcal{B} = \texttt{intent}(\mathcal{C})$*.*

Considering an order relationship defined through inclusion of intents, one may define a *Galois lattice* or *concept lattice*:

Definition 6 *The complete lattice* $\mathcal{L}(\mathcal{K})$ *of concepts of the context* $\mathcal{K}$ *is called (***general***)* **Galois lattice** *or* **concept lattice***.*

Using this formalism, concepts are properly defined and landmarks will be extracted from them.

3.2 Galois Lattice Building Algorithms

Two families of lattice building algorithms exist: incremental algorithms and non-incremental algorithms. Incremental algorithms (*Norris* [Norris, 1978], *Godin* *et al.* [Godin et al., 1991], *Carpineto & Romano* [Carpineto and Romano, 1996], ...) expand the lattice as a new object appears, whereas non-incremental algorithms (*Chein* [Chein, 1969], *Ganter* [Ganter, 1984], *Bordat* [Bordat, 1986], ...) build the lattice after the mapping is filled up.

All these algorithms build a full (or join semi-) lattice, without considering any specificity. The complexity of the algorithms is exponential *w.r.t.* the size of the context, but for some algorithms is linear in the number of concepts modulo some polynomial of the input size [Kuznetsov and Obiedkov, 2002]. Many techniques have been developed with the aim to reduce the complexity, as in [Nguifo and Njiwoua, 1998] where properties are eliminated from the mapping, with an entropy function.

A very complete comparison is done in [Kuznetsov and Obiedkov, 2001] where authors expose algorithms, advantages and drawbacks of each of them.

In our application, images arrive as one goes along. Therefore we have decided to use the Norris algorithm, that is exposed section 6.3, which has been slightly modified in our activity.

4 Formal Landmark Definitions Using Galois Lattices

4.1 General Definition

The context in our application being defined with a set of images (objects), a set of attributes, and a mapping (the presence or not of a attribute f in an image i), the general lattice is built and landmarks are extracted thanks to the following definition:

Definition 7 *Given a context* $\mathcal{K} = (\mathcal{O}, \mathcal{F}, \zeta)$ *and a subset of objects* $\mathcal{A} \subset \mathcal{O}$*. A subset* $\mathcal{B} \subset \mathcal{F}$ *is said to be a* **landmark** *of* $\mathcal{A}$ *if and only if*

– $\mathcal{B}'' = \mathcal{B}$*,*

– *and* $\mathcal{B}' \subseteq \mathcal{A}$*.*

Let us note that our definition of a landmark is equivalent to the definition of a **hypothesis** in [Finn, 1983, Ganter and Kuznetsov, 2000].

By this way, a landmark is a combination of attributes of a concept (intent) that describes a set of images (extent) belonging to a specific place. The complete process, using a *heritage lattice* is wholly described and detailed section 6.

4.2 Landmark Formal Definitions in a Partitioned Context

Let us expose here some definitions useful in our particular context.

4.2.1 Site Landmark Definition

We suppose here to have a context $\mathcal{K} = (\mathcal{O}, \mathcal{F}, \zeta)$ and a partition $(\mathcal{O}_\theta)_{\theta \in \Theta}$ of the object set. So We have $\mathcal{O} = \bigoplus_{(\theta \in \Theta)} \theta$.

Definition 8 *θ is called a* **site** *and Θ the set of sites.*

More generally, a site is a class of objects, in a general classification context.

Definition 9 *$\mathcal{B}_\theta \subset \mathcal{F}$ is said to be a* **landmark** *of a site θ if and only if*

- $\mathcal{B}_\theta^{''} = \mathcal{B}_\theta$,
- *and* $\mathcal{B}_\theta^{'} \subseteq \theta$.

A landmark is thus a set of attributes for which the simultaneous presence is effective in some image of the site to characterize.

4.2.2 Full Landmarks

In particular, if the landmark $\mathcal{B}_\theta$ is a set of attributes present simultaneously in *all* images of the site, $\mathcal{B}_\theta$ is called full landmark.

Definition 10 *$\mathcal{B}_\theta \subset \mathcal{F}$ is said to be a* **full landmark** *of a site θ if and only if*

- $\mathcal{B}_\theta^{''} = \mathcal{B}_\theta$,
- *and* $\mathcal{B}_\theta^{'} = \theta$.

Let us note that full landmarks (also called **unique hypothesis** in [Finn, 1983]) usually never exist in a practical application.

4.2.3 Maximal Landmark and Coverage

If there is no full landmark, it is interesting to limit the number of landmarks by selecting the more general ones; that is why maximal landmarks are introduced.

Definition 11 *A* **maximal landmark** *$\hat{\mathcal{B}}$ in a set of landmarks is a landmark of minimal intent.*

Note that maximal landmarks are equivalent to **minimal hypothesis** exposed in [Ganter and Kuznetsov, 2000]

In a practical way, only maximal landmarks (issued from maximal concept) will be stored during the learning phase.

The coverage of a site by a landmark or a set of landmarks specifies if all images of the site include some of landmarks or not.

Definition 12 *let* $\{\mathcal{B}_{\theta,i}\}_{i=1...N_\theta}$ *be the* N_θ *landmarks of a site* θ*. This site is said to be* **covered***, or the landmarks* **cover** *the site, if and only if*

$$\bigcup_{i=1...N_\theta} \{\mathcal{B}'_{\theta,i}\} = \theta$$

If there is a full landmark in a site, the coverage is obvious. If not, this means that some pictures could not be covered by a landmark. Let us note that the coverage, if it exists, is done by maximal landmarks.

4.3 From Classical Landmarks to Probabilistic Landmarks

Such "classical" landmarks allow to select combinations of attributes that label the subset $\mathcal{A}$. However, most of the case, this too strict definition of landmark leads to a very important "no-response" rate. Indeed, in a real case, a lot of new images will have *a posteriori* no landmark, and thus could not be classified.

That is why we here introduce *probabilistic landmarks*, that will give a non-binary information but a probability to be in each place.

5 Probabilistic Landmarks

Probabilistic landmarks give probabilistic information on the place -in a topological sense- the robot locate itself. From now, landmarks are only defined as a combination of attributes present in one place (*site*) and not in another. However, such information could be expanded with probabilities to improve efficiency, mostly in the case of "no answer".

Let us expose in this section the complete formalism introducing probabilistic landmarks. First, we define landmarks in a partitioned context; second, we define a *probabilistic Galois lattice* and, finally, we will be able to define *probabilistic landmarks* in our application context.

5.1 Probabilistic Galois Lattice

Let us define first *probabilistic concepts*. A concept, as exposed in section 3, is an association of a set $\mathcal{A}$ of objects (extent) and a set $\mathcal{B}$ of attributes (intent): $\mathcal{C} = (\mathcal{A}, \mathcal{B})$. However, in our robotic application, objects (images) have no interest by themselves, but the number of objects that have such and such attribute. That is why we here propose to replace the extent by the probability of having the attribute:

Definition 13 *Let* $\mathcal{C} = (\mathcal{A}, \mathcal{B})$ *be any concept of a lattice* $\mathcal{L}$*. The corresponding* **probabilistic concept***,* $\mathcal{C}^p$*, is the association of a set of attributes* $\mathcal{B}$ *and the probability associated to this set* $p(\mathcal{B})$*.*

Thus we have:

$$\mathcal{C} = (\mathcal{A}, \mathcal{B}) \Rightarrow \mathcal{C}^p = (p(\mathcal{B}), \mathcal{B}) \quad (3)$$

The probability associated to a subset $\mathcal{B}$ in a concept is simply deducted from the number of objects having $\mathcal{B}$ among attributes:

$$\mathcal{C} = (\mathcal{A}, \mathcal{B}) \Rightarrow p(\mathcal{B}) = \frac{card(\mathcal{A})}{card(\mathcal{O})} \quad (4)$$

5.2 Probabilistic Landmarks

Let us transpose this in a partitioned context. The probability of having one particular extent in a general context has no meaning, but that of having one particular extent in **each** site is important and useful.

Let $\mathcal{B} \subset \mathcal{F}$ be a combination of attributes and θ a specified site. The goal is thus, "seeing" $\mathcal{B}$, to find the probability to be in θ: $p(\theta|\mathcal{B})$.

The following formula allows us to establish such a experimental probability:

$$p(\theta|\mathcal{B}) = \frac{p(\theta \cap \mathcal{B})}{p(\mathcal{B})} \quad (5)$$

However, this probability is actually "legible" inside concepts through the cardinality of elements:

$$p(\theta|\mathcal{B}) = \frac{card(\mathcal{A} \cap \theta)}{card(\mathcal{A})} \quad (6)$$

Therefore, giving a concept $\mathcal{C} = (\mathcal{A}, \mathcal{B})$ inside a lattice, it is possible to have the probability of being in each site, with a particular combination of attributes:

$$\mathcal{C}^p = ((p(\theta_1|\mathcal{B}), (p(\theta_2|\mathcal{B}), \ldots, (p(\theta_{N_s}|\mathcal{B})), \mathcal{B}) \quad (7)$$

where N_s is the number of sites.

5.3 Classical Landmarks and Probabilistic Landmarks

As defined earlier, a classical landmark is a set of attributes that is present in one site and absent in all other sites. Probabilistic landmarks, on the other hand, do exist everywhere, but with their intrinsic probabilities.

The relationship between a classical landmark and a probabilistic landmark is thus the value of the probabilities. Using the strict definition of a landmark, a probabilistic landmark attached to a site θ is a landmark of θ *if and only if* the probability is equal to one:

$$\mathcal{B}_\theta \text{ landmark of } \theta \Leftrightarrow p(\theta|\mathcal{B}_\theta) = 1 \quad (8)$$

5.4 Comparison with Other Galois Lattice-Based Classification Systems

Many lattice-based classification systems exist. See [Nguifo and Njiwoua, 2003] for a complete description, and [Kuznetsov, 2004] for review of a lattice-based Machine Learning.

RULEARNER [Sahami, 1995] is a very closed lattice-based system using classical landmarks. However, though our landmark definition is similar, the selection process is very different in that we use heritage lattices in our application. Advantages of such lattices is shown in section 6.

Now, the closest system using "probabilistic landmarks" is the GALOIS system [Carpineto and Romano, 1993], which uses a similarity based on the number of common attributes.

Nevertheless, in our application, the similarity measure is relative to the number of attributes present in both the concept and the object. Thus a set of attributes could be very useful and conclusive even if its cardinality is small.

Moreover, finding a similar concept is not enough to validate the classification: the experimental probability must be above a threshold T we have fixed arbitrary.

Finally, the full classification process uses both classical landmarks and probabilistic landmarks, if needed.

6 Landmarks Selection Process

Here is presented the general Galois lattice-based landmark selection algorithm, that uses the heritage lattice, followed by the new probabilistic landmarks-based algorithm.

6.1 Classical Landmark Based Algorithms

Following the strict definition of a landmark, the general lattice is built and concepts are put into a hierarchy. Considering all concepts $\mathcal{C}_\theta$ relative to a site θ, *i.e.* all concepts whose extent are (only) some images of the place, landmarks are intents of these concepts. The algorithm is given Figure 4.

However, in a practical way, *heritage lattices* [Godin et al., 1995] use *H-concepts* to avoid attribute redundancy. Let us formalize them:

Definition 14 *Let $\mathcal{L}$ be a concept lattice and $\mathcal{C}$ any node of this lattice. The* **H-concept** $\tilde{\mathcal{C}} = (\tilde{\mathcal{A}}, \tilde{\mathcal{B}})$ *corresponding to the concept* $\mathcal{C} = (\mathcal{A}, \mathcal{B})$ *is the pair (new object(s), new attribute(s)), new objects (resp. new attributes) being w.r.t. the child (resp. parent) nodes of C.* $\tilde{\mathcal{A}}$ *and* $\tilde{\mathcal{B}}$ *are called respectively* **H-extent** *and* **H-intent**.

1.	Extract primitives from each image
2.	Determine the presence or not of attributes and fill up the mapping
3.	Build the corresponding general lattice
4.	Select concepts relative to a site θ, *i.e.* concepts whose extents contain only images of the site
5.	Landmarks are intents of such concepts

Figure 4: The classical Landmark Based Algorithm

Let us notice that, generally, H-concepts are not concepts.

Definition 15 *Let $\mathcal{L}$ be a concept lattice. The* **heritage lattice** $\tilde{\mathcal{L}}$ *is the lattice $\mathcal{L}$ in which concepts have been replaced by corresponding H-concepts.*

Considering, in the heritage lattice, the set of H-concepts $\{\tilde{\mathcal{C}_\theta}\}$ relative to a site θ in the heritage lattice, it is necessary to select H-concepts:

1. that are relative to the site θ;
2. that are parents or children of all H-concepts of $\{\tilde{\mathcal{C}_\theta}\}$;
3. and, among them, select the set of H-concepts that have in the H-extent no object related to another site $\theta^* \neq \theta$.

Landmarks are H-intents of those selected H-concepts.

However, in some particular cases, no H-concept is selected for a site θ, so the heritage lattice fails to give landmarks. In such a case, the concept lattice is used to read landmarks in the set of concepts $\{\mathcal{C}_\theta\}$. The general algorithm is presented in Figure 5.

A simple but explicit example is shown in Figure 6.

The cardinalities of H-intents are considerably smaller. This property is very useful in our application. Indeed, an image could be characterized by several hundreds of attributes. Thus reducing the size of concepts means that much less memory is being used by the computer.

6.2 Probabilistic Landmark Based Algorithm

In such a case, all concepts give probabilistic landmarks on each site of the environment. Therefore, in such an algorithm, all concepts are kept, and there is no need to have now any selection process.

1. Extract primitives from each image
2. Determine the presence or not of attributes and fill up the mapping
3. Build the corresponding heritage lattice
4. For each place θ:
41. Select the set of H-concepts $\{\tilde{\mathcal{C}_\theta}\}$ whose H-extents are only images of the place θ
42. Select all ascendants and descendants of H-concepts of $\{\tilde{\mathcal{C}_\theta}\}$
43. Remove H-concepts that have, as descendant, one H-concept including object(s) from another site in the H-extent
44. Eventually, if desired to find maximal landmarks, among remaining H-concepts, remove those that have any other H-concept as ascendant in the same group
45. Finally, read the landmarks in the H-intent of remaining H-concepts. If empty, read the intent of the corresponding concept in $\mathcal{L}$

Figure 5: The Galois Lattice-Based Landmark Extraction Algorithm using a Heritage Lattice

The number of probabilistic landmarks being more important than the number of classical ones, we do have here a richer information to allow a robot to make a decision.

The algorithm is presented in Figure 7.

In a real process, the two last steps carry out together: indeed, probabilities are updated by an algorithm of Norris for building a lattice, which has been slightly modified.

6.3 Modified Norris Algorithm

We implemented the algorithm of Norris [Norris, 1978], which is one of those described in [Kuznetsov and Obiedkov, 2001]. The modification we have added concerns the way concepts update probabilistic landmarks of the lattice. The algorithm is shown in Figure 8.

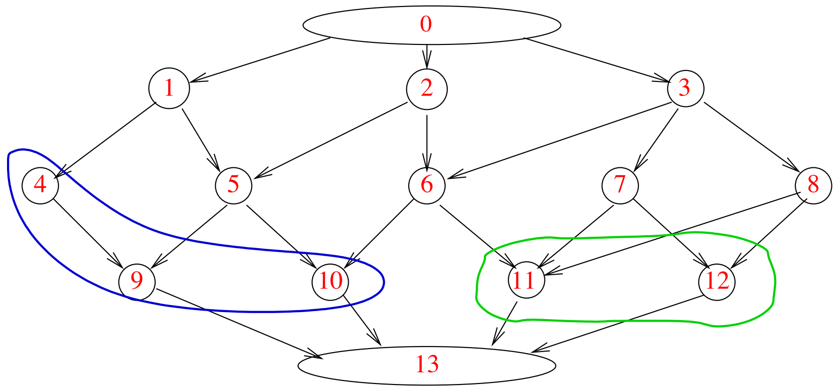


Figure 6: Simple example of landmark extraction. Let us consider here two places and the set of H-concepts related to each place (H-concepts 4, 9 and 10 for the first one and 11 and 12 for the second one). Landmarks of the first place are H-intents (if not empty) of H-concepts 4, 9, 10, and 5, that are ascendants of H-concepts 9 and 10; full landmark of the first place is H-intent (if not empty) of H-concept 1. Note that H-concepts 0, 2 and 6 are not considered for the first place, because they have H-concepts of the second place as descendants. H-concepts of the second place are H-intents (if not empty) of H-concepts 7, 8, 11 and 12, and there is no full landmark for the second place.

---

1. Extract primitives from each image

2. Determine the presence or not of attributes and fill up the mapping

3. Build the corresponding general lattice

4. Determine the probabilistic landmarks

Figure 7: The Probabilistic Landmark Based Algorithm

7 Localization Using Visual Landmarks

Autonomous localization is now fully dependent on visual landmarks. Given every new image $\mathcal{I}$ from the structured environment, landmarks will allow the robot to locate itself. First, the robot tries to find classical landmarks. But if it fails, it will try to find probabilistic landmarks.

NORRIS:

1.	$\mathcal{L} := \emptyset$
2.	For each new object $o \in \mathcal{O}$
2.1	ADD(o,$\mathcal{L}$)
3.	$\mathcal{L}$ is the concept set.

ADD (o,$\mathcal{L}$):

1.	For each $(\mathcal{A}, \mathcal{B}) \in \mathcal{L}$
1.1.	If $\mathcal{B} \subseteq \{o\}'$
1.1.1.	$\mathcal{A} := \mathcal{A} \cup \{o\}$
1.1.2.	$\forall \theta \in \Theta, p(\mathcal{B}\vert\theta) := (card(\mathcal{A} \cap \theta)/card(\mathcal{A}))$
1.2.	Else
1.2.1.	$\mathcal{D} := \mathcal{B} \cap \{g\}'$
1.2.2.	If $\{h\vert(h \in \mathcal{O})\&(h$ has already been added$)\&(\mathcal{D} \subseteq \{h\}')\} = \emptyset$
1.2.2.1.	$\mathcal{L} := \mathcal{L} \cup (\mathcal{A} \cup \{o\}, \mathcal{D})$
1.2.2.2.	$\forall \theta \in \Theta, p(\mathcal{B}\vert\theta) := (card(\mathcal{A} \cap \theta)/card(\mathcal{A}))$
2.	If $\{h\vert(h \in \mathcal{O})\&(h$ has already been added$)\&(\{o\}' \subseteq \{h\}')\} = \emptyset$
2.1.	$\mathcal{L} := \mathcal{L} \cup (\{o\}, \{o\}')$
2.2.	$p(\mathcal{B}\vert\theta = \Theta(o)) := 1$
2.3.	$\forall \theta^* \neq \Theta(o), p(\mathcal{B}\vert\theta^*) := 0$

Figure 8: Modified Norris Algorithm

7.1 Localization Using Classical Landmarks

The rule of localization of classical landmark is quite very simple:

- if there is, in the new image $\mathcal{I}$, one or several landmarks relative to one and only one site, the robot is in this site;
- if there is no landmark, the robot cannot locate itself;
- if there are landmarks from at least two different sites, the lattice is updated with the new image. Once this new image is located, landmarks are updated.

7.2 Localization Using Probabilistic Landmarks

The rule is here quite different than earlier. First, what is the most similar concept is needed.

Definition 16 *Let $\mathcal{C}_1 = (\mathcal{A}_1, \mathcal{B}_1)$ and $\mathcal{C}_2 = (\mathcal{A}_2, \mathcal{B}_2)$ be two arbitrary concepts, the similarity $\mathcal{S}(\mathcal{C}_1, \mathcal{C}_2)$ is defined by the formula:*

$$\mathcal{S}(\mathcal{C}_1, \mathcal{C}_2) = \frac{card(\mathcal{B}_1 \cap \mathcal{B}_2)}{card(\mathcal{B}_1 \cup \mathcal{B}_2)} \tag{9}$$

Let $\mathcal{B}_\mathcal{I} \subset \mathcal{F}$ be the set of attributes in the new image $\mathcal{I}$. The most similar concept $\mathcal{C}^*$ of the new image associated to its attributes is thus defined by:

$$\mathcal{C}^* = \max_{\mathcal{C} \in \mathcal{L}}(\mathcal{S}(\mathcal{C}, (\mathcal{I}, \mathcal{B}_\mathcal{I}))) \tag{10}$$

Secondly, once the most similar concept found, the localization in given by the probability of being in a site. The robot is in the site θ^* defined by:

$$\theta^* = \max_{\theta}(p(\theta|\mathcal{B}^*)) \tag{11}$$

where $\mathcal{B}^*$ is the intent of $\mathcal{C}^*$.

In our application, in order to have a security margin, we have introduced a threshold:

$$\theta^* \text{ selected } \Leftrightarrow p(\theta^*|\mathcal{B}^*) > T = 0.75 \tag{12}$$

7.3 Localization Algorithm

Figure 9 exposes the general localization algorithm. If the robot does not detect any (full or not) landmark, it will try to locate itself using probabilistic landmarks.

8 Experimentation and Results

Experimentation have been done is the SUPAERO laboratory, with the robot *Pekee* (Fig 10) equipped with a CCD color camera, in a real four-place (-site) structured environment.

177 images have been taken for the learning stage, in four different places of the laboratory environment. The attribute extraction process gives a 177×66 mapping. The corresponding 5265 concept lattice is computed in 8 seconds on an AMD Athlon 2400+ machine. For the four classes, 883 concept-landmarks are extracted, and 42 maximal landmarks are kept: 9 for the first place, 8 for the second one, 17 for the third one, and 8 for the fourth one (see Table 1).

8.1 Results with Classical Landmarks

During the generalization phase, 32 images are issued from the place #1. These images are *different* from those of the learning phase. Landmarks are searched

As soon as a new image is captured:

0.	The robot is not located
1.	Analyse the image and look for attributes
2.	Find landmark(s) among attributes
2.1.	If there is landmarks from one and only one site
2.1.1.	The robot is located
2.2.	Else if there is landmarks from different sites
2.2.1.	The robot cannot locate
2.3.	Else if there is no landmark
2.3.1.	Select the most similar concept
2.3.2.	If the highest probability is above a threshold T
2.3.2.1.	The robot is located
3.	If the robot is not located
3.1	Move to capture another picture

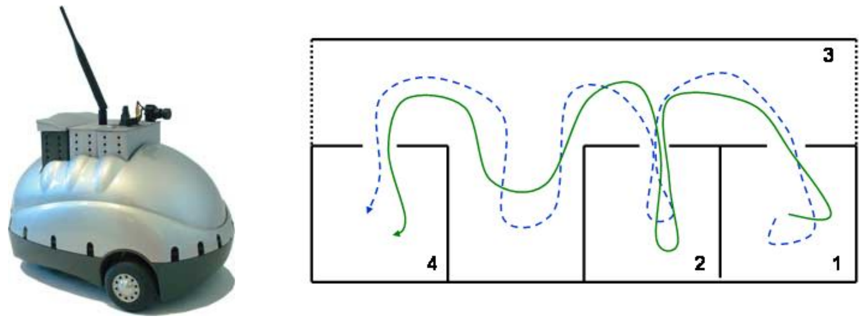
Figure 9: General Localization Algorithm

Place	**Full Landmarks**	**Landmarks**	**Maximal Landmarks**
Place #1	0	194	9
Place #2	0	316	8
Place #3	0	291	17
Place #4	0	82	8
Total :	0	883	42

Table 1: Landmarks Extraction.

on all images: 1 image contains 2 ambiguous landmarks (one of the place #1, one of the place #3) and 14 no landmark; 16 images contain only landmarks of the place #1, and 1 image contains a place #4 landmark. There is thus a response rate of 53.1%, an absolute well situated image rate of 50% on all images, more important a relative well situated image rate of 94.1% on (well or not) located images, an absolute error rate of 3.1% and a relative error rate of 5.88%. Identical analysis has been done for the four places, Table 2 (NI: Number of Images, NR: Number of Responses, NGR: Number of Good Responses, NFR: Number of False Responses).

Let us note that the generalization phase has been processed firstly with



(a) (b)

Figure 10: Pekee with its color camera and the map of the structured environment.

Place	NI	NR	NGR	NFR
Place #1	32	17 (53.1%)	16 (94.1%)	1 (5.88%)
Place #2	50	13 (26%)	12 (92.3%)	1 (15%)
Place #3	31	10 (32.3%)	10 (100%)	0 (0%)
Place #4	38	20 (52.6%)	19 (95%)	1 (5%)
Total :	151	60 (39.8%)	57 (95%)	**3 (5%)**

Table 2: Results using Classical Landmarks

the learning set of images. Of course, for each place, there is no landmark from another place, but also the response rate is not 100% (88%, 43.1%, 85.7% and 54.8% for respectively place #1, #2, #3 and #4): there is some images with *a posteriori* no useful information, *i.e.* images whose attributes are shared with some pictures of other sets.

8.2 Results with an Optimized Neural Network

A classical neural network under MatLab has been processed to validate our approach. To be optimized, several experimentations have been computed to obtain the best network as possible.

The optimized network is composed of 66 neurons in the first layer (corresponding to our 66 attributes), 66 neurons in the middle layer and 4 neurons (corresponding to the 4 places) in the last layer. The training function is a Backpropagation gradient training with an adaptive learning (`taingda`), with a hyperbolic tangent sigmoid transfert function for each layer of the network. Other comparisons have been done with different number of layers, different number of neurons in the middle layer, different training process and/or different transfer functions, but with worse results. The Levenberg-Marquardt and Bayesian regularization algorithms fail due to the high number of entries.

With the number of 700 training epochs, the smallest learning rate is 4.10^{-2} and more significantly the smallest error rate (false response compared to all response) we obtained is 5% on the training set of images, and 30% on the test set...

Moreover, the variability of responses of a network is very different from one learning process to another, with the same training database. Best results cited above are reached once on five or six tries.

To fit with our technique and to have comparable results (see Table 3), a program has been developed to allow the neural network to give some "no-responses". In a practical way, the classification answer is validated if and only if the difference between the greatest probability to be in one place and the second greatest probability to be in another place is above a threshold, that is adjusted to have the same rate of no-responses.

Place	**NI**	**NR**	**NGR**	**NFR**
Place #1	32	17 (53.1%)	17 (100%)	0 (0%)
Place #2	50	17 (26%)	17 (100%)	0 (0%)
Place #3	31	14 (45.2%)	10 (71.4%)	4 (28.6%)
Place #4	38	12 (31.6%)	10 (83.3%)	2 (16.6%)
Total :	151	60	54	**6 (10%)**

Table 3: Results using a Neural Network.

8.3 Results with Probabilistic Landmarks

Results we have obtained with probabilistic landmarks are quite better than classical landmarks (Table 4).

Even if the final result is quite identical in terms of good response rate (94.5% against 95%), the number of well-located images is higher (73 against 60, *i.e.* 21.7% much better !!), which was the goal of introducing probabilistic landmarks.

Place	**NI**	**NR**	**NGR**	**NFR**
Place #1	32	25 (78.1%)	24 (96%)	1 (4%)
Place #2	50	14 (28%)	11 (71.4%)	3 (28.6%)
Place #3	31	12 (38.7%)	12 (100%)	0 (0%)
Place #4	38	22 (57.9%)	22 (100%)	0 (0%)
Total :	151	73 (48.3%)	69 (94.5%)	**4 (5.5%)**

Table 4: Results using Probabilistic Landmarks

9 Conclusion and Perspective

Our original approach using Galois lattices gives very good results compared to classical techniques such as neural network for selecting visual landmarks. Moreover, the initial technique using classical landmarks, giving a high rate of good results but with a weak rate of answers, is improved by the introduction of probabilistic landmarks. Such landmarks give a probability for the robot to be in each place of the structured environment through the probabilistic concepts inside the lattice it has built.

The next step is to implement active vision for our robot: now, the robot just "waits" to know where it is, and results will be improved a lot by introducing a landmark searching algorithm if the robot does not know where it is.

Finally, a semi-supervised learning is about to be implemented, where a robot detects room changes without knowing whether it is in a new place or not.

References

[Arleo et al., 2001] Arleo, A., Smeraldi, F., Hug, S., and Gerstner, W. (2001). Place cells and spatial navigation based on 2d visual feature extraction, path integration, and reinforcement learning. *Advances in Neural Information Processing Systems 13, MIT-Press.*

[Barbut and Monjardet, 1970] Barbut, M. and Monjardet, B. (1970). *Ordre et Classification.* Hachette Université.

[Bordat, 1986] Bordat, J. (1986). Calcul pratique du treillis de galois d'une correspondance. *Mathématique, Informatique et Sciences Humaines.*

[Carpineto and Romano, 1993] Carpineto, C. and Romano, G. (1993). Galois: An order-theoric approach to conceptual clustering. In *Proceedings of ICML'93*, pages 33–40.

[Carpineto and Romano, 1996] Carpineto, C. and Romano, G. (1996). A lattice conceptual clustering system and its applications to browsing retrieval. In *Machine Learning*, volume 24, pages 95–122.

[Chein, 1969] Chein, M. (1969). Algorithme de recherche des sous-matrices premières d'une matrice. *Bulletin Mathématique de la Sociologie Scientifique de la R.S. de Roumanie.*

[Finn, 1983] Finn, V. K. (1983). On machine-oriented formalization of plausible reasoning in the style of F. Bacon - J.S. Mill. In *Semiotika i Informatika*, volume 20, pages 35–101.

[Ganter, 1984] Ganter, B. (1984). Two basics algorithms in concept analysis. Technical report, Technische Hochschule Darmstadt.

[Ganter and Kuznetsov, 2000] Ganter, B. and Kuznetsov, S. (2000). Formalizing hypotheses with concepts. In *(ICCS'2000), LNCS/LNAI*, pages 342–356.

[Ganter and Wille, 1999] Ganter, B. and Wille, R. (1999). *Formal Concept Analysis*. Springer-Verlag.

[Godin et al., 1995] Godin, R., Mineau, G., and Missaoui, R. (1995). Incremental concept formation algorithms based on galois (concept) lattices. *Computational Intelligence, vol 11*.

[Godin et al., 1991] Godin, R., Missaoui, R., and Alaoui, H. (1991). Learning algorithms using a galois lattice structure. *IEEE Int. Conf. on Tools for AI*.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151.

[Hayet et al., 2002] Hayet, J., Lerasle, F., and Devy, M. (2002). A visual landmark framework for indoor mobile robot navigation. *Int. Conf. on Robotics and Automation*.

[Itti et al., 1998] Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259.

[Knapek et al., 2000] Knapek, M., Oropeza, R.-O., and Kriegman, D. (2000). Selecting promising landmarks. *IEEE Int. Conf. on Robotics and Automation*.

[Kuznetsov, 2004] Kuznetsov, S. (2004). Machine learning and formal concept analysis. In *LNCS/LNAI, (ICFCA 2004), vol 2961*, pages 287–312.

[Kuznetsov and Obiedkov, 2001] Kuznetsov, S. and Obiedkov, S. (2001). Algorithms for the construction of concept lattices and their diagram graphs. *Principles of Data Mining and Knowledge Discovery (PKDD 2001), Freiburg, Germany*.

[Kuznetsov and Obiedkov, 2002] Kuznetsov, S. and Obiedkov, S. (2002). Comparing performance of algorithms for generating concept lattices. *Journal of Experimental and Theorical Artificial Intelligence*, 14(2-3).

[Mata et al., 2001] Mata, M., Armingol, J. M., de la Escalera, A., and Salichs, M. (2001). A visual landmark recognition system for topological navigation of mobile robots. *IEEE International Conference on Robotics and Automation (ICRA01)*.

[Nguifo and Njiwoua, 1998] Nguifo, E. M. and Njiwoua, P. (1998). Using lattice-based framework as a tool for feature extraction. In *European Conference on Machine Learning*, pages 304–309.

[Nguifo and Njiwoua, 2003] Nguifo, E. M. and Njiwoua, P. (2003). Supervised classification and formal concept analysis. *International Conference on Formal Concept Analysis, Technische Universität Darmstadt*.

[Norris, 1978] Norris, E. (1978). An algorithm for computing the maximal rectangles in a binary relation. *Revue Roumaine de Mathématiques Pures et Appliquées 23 (2)*.

[Sahami, 1995] Sahami, M. (1995). Learning classification rules using lattices. In *Proceedings of ECML'95*, pages 343–346.

[Thrun, 2002] Thrun, S. (2002). Robotic mapping: A survey. In Lakemeyer, G. and Nebel, B., editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann.

[Thrun et al., 2000] Thrun, S., Fox, D., Burgard, W., and Dellaert, F. (2000). Robust monte carlo localization for mobile robots. *Artificiel Intelligence*.

[Zenou and Samuelides, 2003] Zenou, E. and Samuelides, M. (2003). Characterization of image sets : The galois lattice approach. *International Conference on Advanced Concepts on Intelligent Vision Systems, Ghent, Belgium*.