Aided Architectural Sketching with Markov Models

Dromies and Recognition

Guéna François[1], Lecourtois Caroline[2]
[1,2]Ariam-larea, Ecole nationale supérieure d'architecture de Paris-la-Villette, France.
www.ariam-larea.archi.fr
[1]francois.guena@ariam-larea.archi.fr,
[2]caroline.lecourtois@ariam-larea.archi.fr

Abstract: *This article presents the current state of an ongoing research project developed at the ARIAM-LAREA laboratory of the school of Paris La Villette. The aim of this research is to offer architects digital tools that allow them to explore the project with freehand sketches as they do with pencil and paper. The role of these tools is to interpret the freehand sketches in order to work on them and transform them. They produce from these freehand sketches a digital model from which it will be possible to build a 3D representation, or to evaluate the project from any technical point of view. This paper only shows the part of the system that recognizes particular strokes in an architectural drawing.*

Keywords: *Design; digital aid; architecturology; sketch interpretation; Hidden Markov Model.*

Introduction

This article deals with a part of a research on digital tools for architectural design. Developing digital tools to conceive architecture is one of the two axes of the research program of our laboratory Ariam-Larea (the other one affects the knowledge on the cognitive activity of the architectural conception). This research we present here is founded on the hypotheses that architects still need to use pencil and paper at the beginning of a new project study. Nowadays, digital modeling is not really suitable to conceive architecture and remains more useful when the essential parts of the project have been designed. Thus, there is a gap between the early design phases and the rest of the design process. To fill this gap our objective is to offer architects digital tools that allow them to explore the project with freehand sketches, as they do with pencil and paper. The role of these tools is to interpret the freehand sketches in order to work on them and transform them to produce a digital model, from which it will be possible to build a 3D representation or, to evaluate the project from any technical point of view.

In this context, two digital tools for freehand sketching are currently developed: one by F. Guéna et L.P. Untersteller (Guéna and Untersteller, 2008), a strokes recognizer system that can build different projection views of an object from a first incomplete drawn projection. Another one by F. Guéna and C. Lecourtois (Lecourtois and Guéna, 2009) named ESQUAAS, which is the objective of our article.

ESQUAAS (in French "ESQUisse Architecturologiquement ASsisté", architecturological aids for sketch), is a software in progress that will help architects to conceive, by recognizing and interpreting in a 3D model their freehand sketches drawn on a graphical screen. Then ESQUAAS is a digital system of recognition and interpretation that is in favor of the practice of freehand sketches, in architectural design.

This digital system will be composed with two levels of recognition: 1) one that treats strokes and 2), the other one that treats cognitive operations of architectural conception. These two levels of recognition are developed from a scientific knowledge about architectural conception, namely Architecturology and which explains cognitive activity. Architecturology was initiated and constituted by a well-known French Theoretician of architecture, Philippe Boudon (Boudon, 1991; Boudon, 1992; Boudon and al., 1994; in Albertsen, 1999). Its scientific object relates to the manners that architects conceive spaces in giving them relevant measures. We distinguish conception from design, and work on architectural conception as a cognitive activity aiming to create a new architectural space. Meanwhile, design includes all the activities that are implied in architectural projects.

More precisely about our works on Digital Aids, we retain two concepts from Architecturology ("*Dromies*" and "*Scales*"). Each of them helps to think of one of our two levels of recognition.

The concept of "*Dromies*" is built to describe different types of relationships between strokes. They constitute a list of different kinds of these relationships. The concepts determining them were constructed from the Greek term of dromos that means course and/or trace or stroke. This list is named "*Dromies*". "*Dromy*" (singular of "*Dromies*") as an architecturological concept means an elementary unit of a stroke.

The other concept of "*Scales*", relating to cognitive operations of architectural conception, comes from the question of scale in architecture. What is the meaning of a building having a scale? That was the question of the theoreticians of LAREA. From the different answers of this question, they have coined a scientific concept of "*Scale*" ("*Architecturological Scale*") to identify different manners giving measure to architectural space. From this theoretical purpose and from cases of architectural projects, they have separated twenty possible "*scales*" as twenty classes of manners to give measure to space (*Human Scale*, *Parcel Scale*, *Geometrical Scale*, *Geographical Scale*, *Technical Scale*, etc.) (Boudon et al., 1994; in Albertsen, 1999). By extending this theoretical system to Urbanism, a Twenty-First Scale was discovered to complete the initial model: an *Incorporating Scale* (Lecourtois, 2007). Each of these twenty-one "*Scales*" identifies a possible kind of relationship between a project and its complex context. They are all currently used in order to constitute a digitally recognized multi-agent system of cognitive operations of conception (Lecourtois and Guéna, 2009).

This paper only explains the level of Strokes Recognition Mechanism of the system that is based on the Architecturological concept of "*Dromy*" or "*Dromies*". The first section presents a part of the "*Dromies*" list, twelve kinds of binary strokes. In the next section, the digital tool is described as a multi-agent system which memorizes and computes the properties of different kinds of strokes in order to recognize them in freehand sketches.

Dromies: a theoretical model on strokes used in architectural design

The work on the relationships between graphical strokes was led from the production of letter drawings coming from the different schools of architecture (as those of the Bauhaus, Ulm, Amsterdam, De Stijl, etc.). Moreover, the relationship that Le Corbusier has introduced between architecture and typography brought the theoreticians of LAREA to question the graphical syntax of sets of initials. From the analysis of 239 sets of initials, they have constituted a list of "*Dromies*" (Pousin and Boudon, 1988).

Five classes of "*Dromies*" were distinguished relatively to the constitution of the graphical strokes (*Catadromy*, *Monodromy*, *Heterodromy*, *Autodromy* and *Metadromy*). From this list of classes, we have chosen to work on "*Dromies*" that are constituted with two strokes: a referential and a stroke drawn relatively to it. These "*Dromies*" are some of the *Metadromies* that are built with binary relationships between strokes.

The above figure (Figure 1) presents the twelve kinds of "*Dromies*" that we are working on for our software. The twelve kinds of *Metadromies* have been separated by analyzing freehand sketches of Alvar Aalto and of Carlo Scarpa. They point the specificity of Architectural freehand strokes that are to be composed with irregular lines and non geometrical figures.

The *Isodromy* (Figure 1:1) is a binary relationship in which the stroke follows its referential so that it covers it on parts. The *Homodromy* (Figure 1:2) is almost the same but with a stroke that does not cover its referential. The *Homodromy* resembles parallelism. The *Antidromy* (Figure 1:3) is the relationship of opposition between a stroke and its referential. The *Acrodromy* (Figure 1:4) is a discontinuing stroke that follows a virtual referential line. The *Paradromy* (Figure 1:5) is a binary relationship in which a serpentine follows the direction of a referential in cutting it at different points. The *Anadromy* (Figure 1:6) is a stroke that jumps over its referential, like a road over a railway. The *Epidromy* (Figure 1:7) represents a tangent relationship between two traces. The *Peridromy* (Figure 1:8) is a circumscribing of a referential point or stroke. The *Apodromy* (Figure 1:9) is the case of strokes that depart from each other around an obstacle. The *Pseudodromy* (Figure 1:10) is the relationship in which little straight-lines have all the same affinity with a referential direction (S*yndromics*), and are in between *Homodromics* (Figure 1:1). The *Balodromy* (Figure 1:11) are strokes that have the dynamic gesture of drawing for referential. The *Holodromy* (Figure 1:12) is a trace unifying its referential by giving it or them a gestalt shape or by supporting them.

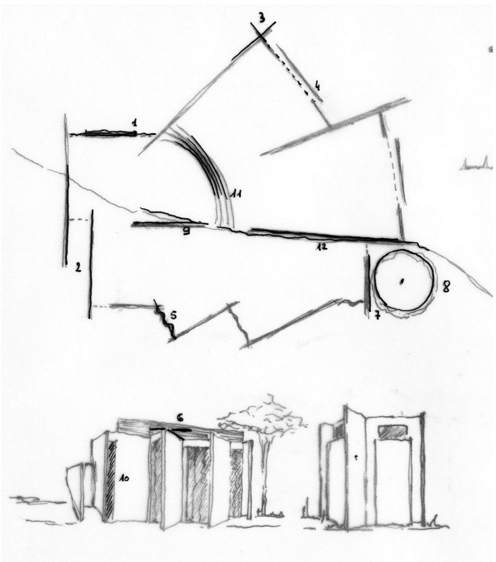


Figure 1
markings of Metadromies on sketches for an utopist project. C. Lecourtois/ 1. Isodromy, 2. Homodromy, 3. Antidromy, 4. Acrodromy, 5. Paradromy, 6. Anadromy, 7. Epidromy, 8. Peridromy, 9. Apodromy, 10. Pseudodromy, 11. Balodromy, 12. Holodromy

Whereas "*Dromies*" are just concepts to identify different relationships between strokes, they are, to us, potential tools to approach architectural conception. Each of these *Metadromies* can be considered as a physical manifestation of cognitive operations that aim to conceive an architectural space. So "*Dromies*" themselves have no real signification but, once coupled with tools to approach the cognition as "*Architecturological Scales*", they can help understanding the activity of architectural conception.

So, our digital recognition system of freehand architectural sketches will be suitable to architectural design if it takes care of these two elements: "*Scales*" and *Metadromies*. As we wrote above, the "*Architecturological scales*" are currently re-questioned by our laboratory in order to transform them into digital multi-agent contexts of recognition and interpretation of cognitive operations of architectural conception. In this future digital environment, *Metadromies* will be mediums of the thought and interfaces of action for future users. Thinking about our digital aids to architectural design as a complexity between these two levels of recognition and interpretation, demands programming one of them and thinking of the manners that it can be augmented, by the second.

The following part of this article explains the mechanism of the implementation of the *metadromies* in a digital recognition system. Digital tools for strokes recognition imply to think of: 1) ways that are possible to memorize and compute them in machine and 2) ways that they can be used to recognize and interpret new strokes in real time. The Hidden Markov Models gives us tools to answer these two requirements.

Hidden Markov Models: theoretical models to memorize "*Dromies*"

For this research we have specifically developed a tool that aims to build and adjust the recognition mechanism. The recognizer is a multi-agent system, composed with several autonomous agents. The graphical interface of the recognizer allows us to draw strokes on a screen or a tablet with a pen. The multi-agent recognizer has to compute properties of the strokes from which it will be possible to identify *Dromies*. Each agent of the system is specialized in identifying a particular *Dromy*.

When the pen is placed down during a stroke drawing the system records 2D points, times and pressures. The system also looks at the elements of the sketch around each sampled points of the stroke. When the stroke is sketched across, over or close enough to entities already present in the drawing, these entities and the corresponding snap modes (near, across, on, end etc.) are recorded in a sequence.

For example, suppose the code for the snap modes are as follows: nothing = 0, end = 1, middle = 2, near = 3, then the user draws a first stroke called "A" on an empty sheet (Figure 2:a) and if the stroke has been sampled in a list of 14 points, the sequence of snap codes for stroke "A" contains a list of 0: Sequence(A)=[0,0,0,0,0,0,0,0,0,0,0,0,0,0]. Suppose the user draws another stroke close to stroke "A" (homodromy) (Figure 2:b). If this stroke, that will be called B, has been sampled in a list of 16 points, the sequence of snap mode could be as follows : Sequence (B)=[1,1,3,3,3,3,2,2,3,3,3,3,3,1,1,0].

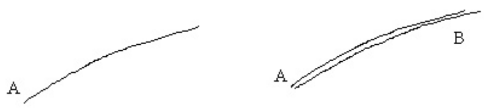


Figure 2
a) (left) a stroke drawn on an empty sheet,
b) (right) a new stroke drawn near the first one

The system can also pick Geometrical data up whilst drawing. For example it can record, in a sequence the succession of variations in directions between two successive sampled points, or the evolution of the distances between two strokes.

The recognition mechanism is founded on Hidden Markov Model (HMM). The HMM is an Artificial Intelligence Technique that deals with ordered sequences of events and imprecise data. Speech and hand-written language recognition tools often use HMM.

A HMM is generally represented by an oriented graph where nodes represent the states and arcs represent the transitions between the states. Each state is associated with probability distributions. A HMM can compute the probability that a sequence of events could have been produced by it.

It is difficult to explain precisely how, in this article, a HMM runs. For further information, we advise the reader to look at the literature of the subject, the Rabiner's paper for example (Rabiner, 1986). We only want to demonstrate here that it is possible to use such HMM for classifying sequences of snap events or geometric properties of strokes and thus recognizing "*Dromies*". Each agent of our system memorizes a HMM with specific probabilities in recognizing a sequence of events associated with one of the "*Dromies*" (Figure 5). When the user performs a

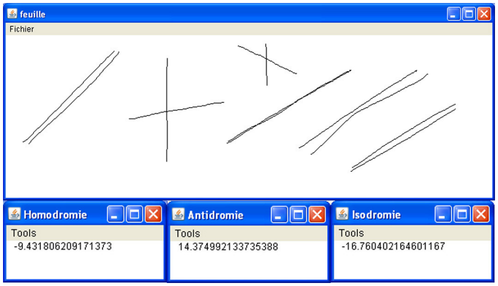


Figure 3
The interface of the system contains a window for the sheet and a window for each agent

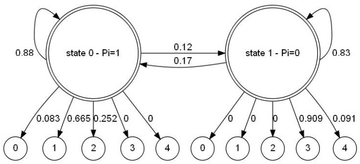


Figure 4
Graphic representation of the HMM associated with the Homodromy agent. The HMM has two states; state0 and state1 and the probability that the sequence begins on the state0 is Pi=1. The probability that the automate stays in state0 for the following events is 0.88. The code for the snap mode is: nothing=0, end=1, middle=2, near=3 and on=4. So the HMM expresses that the first snap events for a sequence in a homodromy are generally "end" snap events (end=1).

stroke each agent computes the probability that this stroke corresponds to its own *Dromy* (Figure 3).

An important property of the HMM is to be computed by learning mechanisms from a set of training sequences. Thus, the system adapts the way it recognizes "*Dromies*" from the manner in which the user draws.

The Java implementation

The software program is developed in Java. We use the open source jahmm library (http://jahmm.googlecode.com/: April 2006) which gives an implementation of HMM related algorithms. This section briefly describes the main classes which compose the Java program. The sheet class defines attributes for memorizing the strokes that the user draws and the methods for managing the pen events. When the pen is down, the system creates a new stroke object. When the pen is down and moved, the system records sequence stroke points, pressures, time and if there is any, the snapped strokes with the corresponding snap modes. Sequences of geometrical properties are also computed during the drawing. The stroke class defines attributes for recording all these sequences.

The agent class implements the Java *running* interface so that each instance of agent runs in its own thread. The agent objects are autonomous. They permanently look at the sheet object and can react as soon as a new stroke is drawn. The agent class defines an attribute which references an instance of an HMM object of the jahmm library. An agent contains methods for learning from training sequences.

The *Dromy class* is an "*abstract class*" in Java language and, there are twelve "*subclasses*" for each kind of "*dromies*". Each subclass memorizes a HMM with two states as shown in figure 4.

Experiments and outcomes

The multi-agent system interface is composed with a Java frame that represents a sheet object in which the user can draw strokes. Around the sheet frame is located other frames that represent the agents. Each agent frame has a menu item for learning. So that the user can draw a set of training sequences for one kind of "*Dromies*" and ask an agent to learn how to recognize this kind of "*dromies*". Figure 3 presents the graphical interface of this system.

Three agents have been currently implemented for the *Homodromy*, *Antidromy*, *Isodromy*. All the HMM probabilities have been computed by a learning mechanism from a set of training sequences.

Figure 5 presents some results for a few experiments. It shows that the results are relatively good even in exploiting only the sequences of snap (line 1 to 6). There are ambiguities between *homodromy* and *isodromy* (line 7). There is an error in recognition in line 8. For improving this system we also need to analyze geometrical properties of strokes: e.g. the evolution of the sign of points of the new stroke, relatively to the equation of the segments in the snapped stroke.

Conclusion

In this article we have described the first level of an interface for a pen-based digital sketching tool. This interface is a multi-agent system able to recognize specific strokes in a sketch. In order to build and adjust the recognition mechanism we have developed an experimental tool based on the Hidden Markov Model. With the HMM it is possible to classify the snap sequences for recognizing different relationships between strokes. This experimental tool allows

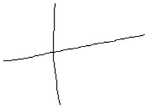
	Homodromy	Antidromy	Isodromy	line
	8.99338235275648	-11.3586780593996	2.668171539476155	1
	-20.8044972703329	30.5491820631485	-35.8615625032779	2
	-infinity	-infinity	13.78482675169832	3
	22.9.829988409351	11.59718243494047	3.2206486174744910	4
	7.341049348669943	-5.68798124543796	0.9099447238531	5
	-9.80561029728006	14.27839180224367	-17.6638969475160	6
	11.98733949140984	-10.8272919353053	9.71464390445999	7
	-10.3798258610995	27.72587949716696	-31.24106759151800	8

Figure 5
Results from experiments. The column presents the score computed by the three agents. The recognized dromy is the one with the best positive score.

us to build different HMMs with learning algorithms from training stroke examples. With these learning experiments it will be possible to attach to each agent of our multi-agent system a specific HMM for recognizing different relationships between strokes in a sketch. To complete the system we have to add new agents for recognizing all the *Dromies*. To improve the recognizing mechanism we also need to analyze the geometrical properties of strokes like variations in direction, distances between strokes etc.

We used the Architecturological concept of "*Dromies"* to identify the different relationships between strokes. Concept of "*Dromies*" is one of the two concepts used to develop the recognition system of a digital aid to conceive that we are developing: ESQUAAS. The second concept is "*Scales*" used to recognize cognitive operations of architectural conception.

We see in this article that, for us, "*Dromies*" and "*Scales*" constitute a design complexity that can clarify the manners that architecture is designed / conceived. We now have to question the different possibilities for each *Metadromy* to have sense in the context of each "*Architecturological Scale*". To begin this investigation, we think that it would be advisable to explore the "*Representational Scale*" at once. "*Representational Scale*" is a class of cognitive operations of architectural conception that includes in design the matter of the architectural representations. It points the fact that the choice of a representational kind, used to conceive a project influences the manner to design the latter. The *Metadromies* which have been explained above as binary relationships between strokes, identify figures of drawings inscribed into a 2D space. The scene of “*Dromies*” is always an orthographic or a perspective projection of the project. According to these two types of projections, the meaning of a *Dromy* can differ. Our work is therefore now to distinguish these cases in order to precisely list the *Metadromies* (to add or separate some) or to identify the possible meanings of them in each type of projection.

References

Albertsen, N. and Lundequist J. (ed.): 1999, Architecturology, Nordisk Arkitekturforskning, Nordic Journal of architectural research 1.99, Göteborg, Författarna och Nordisk Arkitekturforsking.

Boudon, Ph. and Pousin F.: 1988, Figures de la conception architecturale, Dunod, Paris.

Boudon, Ph. (dir.): 1991, De l’architecture à l’épistémologie, la question de l’échelle, PUF, Paris.

Boudon, Ph. Deshayes, Ph. Pousin, F. and Schatz, F.: 1994, reed. 2000, Enseigner la conception architecturale, cours d’architecturologie ,Editions de la Villette, Paris.

Guéna, F. and Untersteller, L.P.: 2008, Computing different projections of a polyhedral scene from a single 2D sketch, in eCAADe 26 conference, Anvers.

Lecourtois, C., 2007 : “Echelles” et Enseignement de la conception, In Les cahiers thématiques de Lille n°6, L’espace de la grande échelle en question, p. .

Lecourtois, C. and Guéna F. : 2009, Eco-conception et Esquisse assistée, in Bignon, J.C, Halin, G. and Kubicki S. (ed.), Conception architecturale numérique et approches environnementales, PUF, Nancy, p.63-75.

Rabiner, L.R. and Juang B.H. : 1986, An introduction to Hidden Markov Models, IEEE ASSP Mag., pp 4-16.