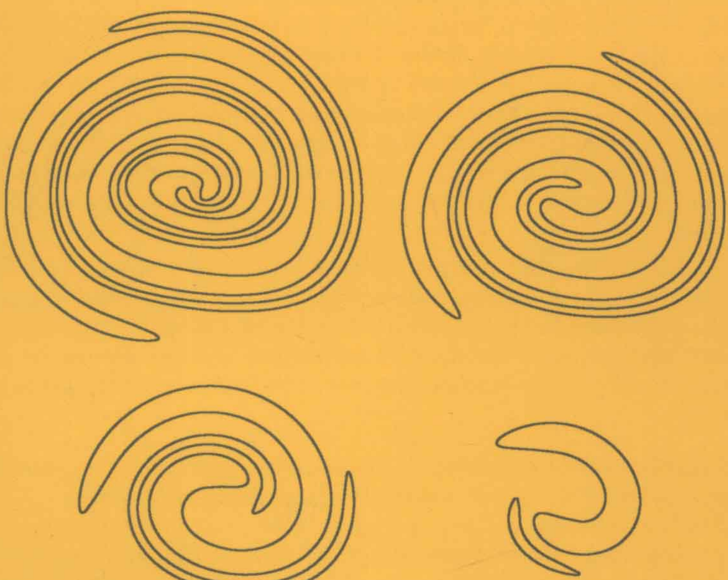
Lecture Notes in Mathematics

Frédéric Cao

# Geometric Curve Evolution and Image Processing

1805



Springer

Frédéric Cao

# Geometric Curve Evolution and Image Processing

Springer

Author

Frédéric Cao

IRISA/ INRIA
Campus Universitaire de Beaulieu
35042 Rennes Cedex
France

*e-mail: Frederic.Cao@irisa.fr*
*http://www.irisa.fr/prive/fcao*

Cataloging-in-Publication Data applied for

Bibliographic information published by Die Deutsche Bibliothek

Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the Internet at http://dnb.ddb.de

Mathematics Subject Classification (2000): 35K65, 35D05, 53A04, 68U10

ISSN 0075-8434
ISBN 3-540-00402-5 Springer-Verlag Berlin Heidelberg New York



Springer-Verlag Berlin Heidelberg New York a member of BertelsmannSpringer Science + Business Media GmbH

http://www.springer.de



Typesetting: Camera-ready TEX output by the author

SPIN: 10904251 41/3142/ du - 543210 - Printed on acid-free paper

# Preface

These lectures intend to give a self-contained exposure of some techniques for computing the evolution of plane curves. The motions of interest are the so-called motions by curvature. They mean that, at any instant, each point of the curve moves with a normal velocity equal to a function of the curvature at this point. This kind of evolution is of some interest in differential geometry, for instance in the problem of minimal surfaces. The interest is not only theoretical since the motions by curvature appear in the modeling of various phenomena as crystal growth, flame propagation and interfaces between phases. More recently, these equations have also appeared in the young field of image processing where they provide an efficient way to smooth curves representing the contours of the objects. This smoothing is a necessary step for image analysis as soon as the analysis uses some local characteristics of the contours. Indeed, natural images are very noisy and differential features are unreliable if one is not careful before computing them. A solution consists in smoothing the curves to eliminate the small oscillations without changing the global shape of the contours. What kind of smoothing is suitable for such a task? The answer shall be given by an axiomatic approach whose conclusions are that the class of admissible motions is reduced to the motions by curvature. Once this is established, the well-posedness of these equations has to be examined. For certain particular motions, this turns to be true but no complete results are available for the general existence of these motions. This problem shall be turned around by introducing a weak notion of solution using the theory of viscosity solutions of partial differential equations (PDE). A complete theory of existence and uniqueness of those equations will be presented, as self-contained as possible. (Only a technical, though important, lemma will be skipped.) The numerical resolution of the motions by curvature is the next topic of interest. After a rapid review of the most commonly used algorithms, a completely different numerical scheme is presented. Its originality is that it satisfies exactly the same invariance properties as the equations of motion by curvature. It is also inconditionally stable and its convergence can be proved in the sense of viscosity solutions. Moreover, it allows to precisely compute motions by curvature, when the normal velocity is a power of the curvature more than 3, or even 10 in some cases, which seems a priori nearly impossible in a numerical point of view. Many numerical experiments are presented.

## Who this volume is addressed to?

We hope that these notes shall interest people from both communities of applied mathematics and image processing. We tried to make them as self-contained as possible. Nevertheless, we skipped the most difficult results since their proof uses techniques that would have led us too far from our main way. Indeed, these lectures are addressed to researchers discovering the common field of mathematics and image processing but also to graduate and PhD students wanting to span a theory from A to Z: from the basic axioms, to mathematical results and numerical applications. The chapters are mostly independent except Chap. 6 that uses results from Chap. 4. The bibliography on every subject we tackle is huge, and we cannot pretend to give exhaustive references on differential geometry, viscosity solutions, mathematical morphology or scale space theory. At the end of most chapters, we give short bibliographical notes detailing in a few words the main steps that produced significant advances in the theory.

## Acknowledgements

My first thanks go to Jean-Michel Morel who was my advisor during my thesis and also proposed me to write these notes. It has been a constant pleasure to work in his team in which collaborations are as numerous as fruitful. I would like to thank him for his scientific support but also for all the other qualities he develops in his relations.
I would also like to express my gratitude to Yves Meyer, with whom any discussion is always an opportunity to learn something and to realize we have still so much to learn. I have also been always impressed by his enthusiasm, and thank him sincerely for his encouragements.
I owe also much to Lionel Moisan, who contributed to the redaction of Chap. 6 and whose help, specially for the numerical part, was very precious.
Many thanks also to Jean-Pierre Le Cadre for his careful reading of this text.
I would also like to thank Andrés Almansa, Vicent Caselles, Francine Catté, Agnès Desolneux, Françoise Dibos, Jacques Froment, Yann Gousseau, Frédéric Guichard, Georges Koepfler, François Malgouyres, Simon Masnou, Pascal Monasse, Pablo Musé, Denis Pasquignon, Frédéric Sur for all the discussions we had for the last few years.

Rennes, October 2002 *Frédéric Cao*

# Contents

## Part II Theoretical curve evolution

# Part I

# The curve smoothing problem

# 1

# Curve evolution and image processing

In this volume, we study some theoretical results and the numerical analysis of the motions of plane curves driven by a function of the curvature. If $C$ is a smooth (say $C^2$) curve, they are described by a partial differential equation (PDE) of the type

$$\frac{\partial C}{\partial t} = G(\kappa)\mathbf{N}, \tag{1.1}$$

where $\kappa$ and $\mathbf{N}$ are the curvature and the normal vector to the curve. This equation means that any point of the curve moves with a velocity which is a function of the curvature of the curve at this point. (See Fig. 1.1.)

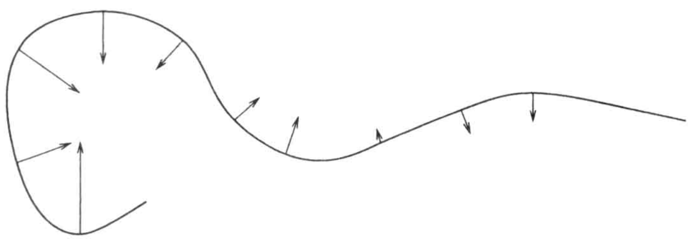


**Fig. 1.1.** Motion of a curve by curvature. The arrows represent the velocity at some points. Here, the velocity is a nondecreasing function of the curvature

These equations appear in differential geometry because the curvature is the variation of the area functional for hypersurfaces. (The length for curves.) In particular, the case $G(\kappa) = \kappa$ can be considered as the gradient flow of the area (length for curve), playing an important role in the theory of minimal surfaces. These equations are also related to the description of crystal growth, where the velocity may also contain an anisotropic term depending on the normal vector. Generally speaking, curvature motions often appear in the motion of interfaces driven by an inner energy or tension, as flame propagation, melting ice, or rolling stone. Surprisingly enough, the motions by curvature have recently appeared in the field of image processing. More precisely, the theory developed in the core of this monograph aims at solving one of the steps that belongs to what has been called low-level vision. It appeared that any automatic interpretation of an image was impossible (or at least very difficult) to perform if one does not apply some preliminary operations on the image. These operations are transformations on the image which make it easier to handle, or simplify it, in order to extract the most basic information more easily. The nature of this information itself is not so easily defined and many researches have tried to mimick the human vision for computational purpose. How vision does really work is still a controversial subject and, except in the next paragraph, we shall not enter into such considerations, but try to remain as practical as possible.

Let us examine a bit closer what an image analysis algorithm should intuitively do. The input of such an algorithm is an image taken by a camera. The output is some interpretation yielding an automatic decision. A commonly accepted method to attain this objective is to detect the objects that are present in the scene and to determine their position and possibly their movement. We could further try to determine the nature of these objects. Before the foundation of the Gestalt School in 1923 [168], it was believed that we detected objects because of the experience we had of them. On the contrary, Gestaltists proved by some psychophysical experimentations that without a priori semantic knowledge, shapes were conspicuous as the result of the collaboration or inhibition of some geometrical laws [99, 98]. Even though the Gestalt laws are rather simple (and were nearly set in mathematical terms by the Gestaltists), their formulation in a computational language is more complex, because they are nonlocal and hierarchically organized. A plan for the computational detection of perceptual information was initiated by Attneave [15], then Lowe [114] and more recently by Desolneux, Moisan and Morel [50]. In fact, most of widely used theories, as edge detection or image segmentation, without strictly following a Gestaltist program, take some part of it into account, since they assume that shapes are homogeneous regions separated from one another and the background by smooth and contrasted boundaries [26, 118, 133, 100], which is in agreement with some grouping Gestalt laws. These theories are often variational and can be formulated with elegant mathematical arguments. (We also refer to a recent book by Aubert and Kornprobst [16] exposing the mathematical substance of these theories.) Very recently, Desolneux, Moisan and Morel [49] developed a new algorithm for shape detection following the Gestalt principles. The advantage of this method is that the edges they found are level lines of images, and consequently, Jordan curves, which are the objects we shall deal with in the following.

We assume (and believe!) that this detection program is realistic but we do not cope with it. On the other hand, this does not mean that we should consider that the problem of shape extraction has been completely elucidated! Nevertheless, as the topic of these lectures follows shape detection, we are obliged to take it for granted.

## 1.1 Shape recognition

Determining automatically the nature of a detected object (is it a man? a vehicle? what kind of vehicle? etc...) is achieved by placing it in some pre-established classification which is the preliminary knowledge. Algorithms use some more or less large databases allowing to precise the classification and try to compare the detected shapes with known ones. Shape recognition is this classification.
Otherwise said, we have a collection of model patterns and we want to know which one the detected shape matches best. A more simple subproblem is to decide whether the observed shape matches a given model. This raises at least two questions:

1. what kind of representation do we take for a shape? (or what is our model of shape?)
2. what kind of properties a shape recognition algorithm should satisfy?

In what follows, we only consider two-dimensional images. The answer to the first question shall be simple: a shape will be a subset of the plane. If the set is regular, it shall be useful to represent it by its boundary. If the set is bounded, its boundary is a closed curve. By the Theorem of Alexandrov 2.4, it is equivalent to know the set or its boundary.
In order to answer the second question, let us follow David Marr in *Vision* [117].

> "Object recognition demands a stable shape description that depends little, if at all, on the view point. This, in turn, means that the pieces and articulation of a shape need to be described not relative to the viewer but relative to a frame of reference based on the shape itself. This has the fascinating implication that a canonical coordinate frame must be set up within the object *before* its shape is described, and there seems to be no way of avoiding this."

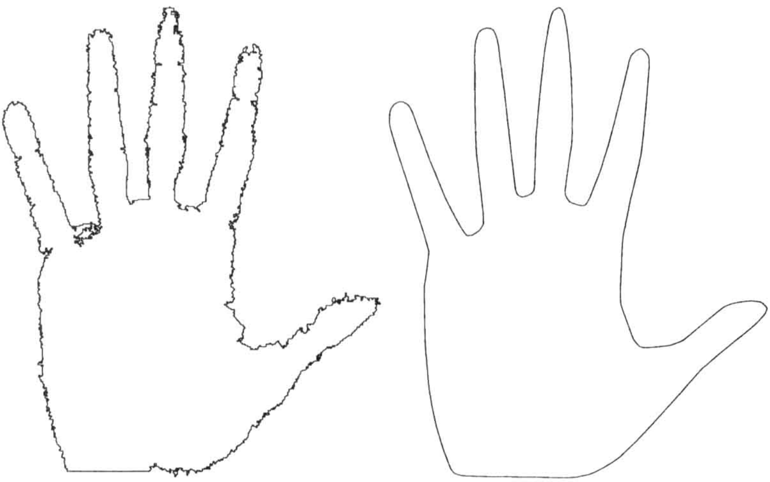
A "canonical coordinate frame" is

> "a coordinate frame uniquely determined by the shape itself."

The description must be stable in the sense that it must be insensitive to noise. For instance, consider the shape given by the curve on Fig. 1.2(a). This curve has been obtained by scanning a hand and then by thresholding the grey level to a suitable value. One has no difficulty to recognize this shape immediately. However, in a computational point of view, this shape is very complicated. A quantitative measure of this complexity is that the curve has about 2000 inflexion points, most of which having no perceptual meaning! Let us now consider the shape on Fig. 1.2(b). This shape has been obtained from the original one by smoothing it with an algorithm described in the following of these lectures. The tiny oscillations have disappeared, and the curve has only 12 inflexion points. In itself, this number has no absolute significance. However, this shape is intuitively better in a computational point of view for three reasons:

1. it is very close to the first one.
2. it is smoother.

3. Fig. 1.2(b) is a good sketch of a hand in the sense that it cannot be much simplified without changing the interpretation. As a parallel, Attneave [15] showed a sketch of a cat containing only a few (carefully chosen) lines, which were sufficient to guess what the drawing was. This means that, up to some point, a shape can be considerably simplified without altering our recognition. In a sense, Fig. 1.2(b) is closer than Fig. 1.2(a) to the minimal description of a hand.



(a) Original shape (b) Smoothed shape

**Fig. 1.2.** The representation of shape for recognition must be as simple and stable as possible. Both shapes represent the same object at different level of details. For a recognition task, most of details on Fig. 1.2(a) are spurious. The shape on Fig. 1.2(b) is intuitively much simpler and visually contains the same information as the noisy one

What about the "canonical coordinate frame"? Mathematicians will call such a frame *intrinsic*. It is not very complicated to imagine such a frame. For instance, the origin may be taken as the center of mass of the shape. Then, the principal directions given by the second order moments provide some intrinsic directions. Those have the nice additional properties to be invariant with respect to rotations. However, if we think of a circle, this intrinsic frame is not uniquely defined. This does not matter since each frame gives the same description, but if we now think of a noisy circle, then the orientation of the frame vector may dramatically change, contradicting the stability hypothesis.

### 1.1.1 Axioms for shape recognition...

Marr did not give precise any practical algorithms for shape representation and recognition but, in a sense, he initiated an axiomatic approach, that has been prolonged by many people. (A recent review with axiomatic arguments is presented by Veltkamp and Hagedoorn [164] for the shape matching problem.) Most people agree that the matching problem is equivalent to finding distances between shapes. (See for example the works of Trouvé and Younes [159, 170].)

#### Intrinsic distance

First, the distance between objects should be independent on the way we describe them. If an object is a set of pixels, it seems clear that a distance taking an arbitrary order of the points into account is not suitable. In the same way, for curves matching, the parameterization should not influence the matching, which should only depend on the geometry of the curves.

#### Invariance

Another property shall also be a cornerstone of our theory: invariance. Marr understood that the recognition should not depend on the particular position of the viewer. The mathematical formulation of invariance is a well known technique, giving extremely important results in many fields such as theoretical physics or mechanics. We consider a set of transformations (homeomorphisms), which has in general a group structure. This group models the set of modifications of the shape when the viewer moves. In a three dimensional world, images are obtained by a projection and such a group does not exist: we cannot retrieve hidden parts of objects by simply deforming the image taken by the camera. However, for "far enough" objects and "small displacements", projective transformations are a good model to describe the modification of the silhouette of the objects since they correspond to the change of vanishing points in a perspective view. If we add some additional hypotheses on the position of the viewer, we can even consider a subgroup of the projective group. If $\mathcal{G}$ is the admissible group of deformations, and $d$ a pseudo-distance between shapes, the invariance property can be formulated by

$$\forall g \in \mathcal{G}, \quad d(gA, B) = f(g)d(A, B), \tag{1.2}$$

where $gA$ represents the shape $A$ deformed by $g$ and $f : \mathcal{G} \to \mathbb{R}$ does not depend on $A$ and $B$. Notice that $d$ is only a pseudo-distance since $d(gA, A) = f(g)d(A, A) = 0$. In a mathematical point of view, it is natural to define shape modulo a transformation, which is equivalent to define a distance between the orbits of the shape under the group action. In such a way, a true distance is retrieved instead of a pseudo-distance.

#### Stability

The stability may be thought as noise insensitivity. In [164], it is formulated by a set of four properties. The first one is strongly related to invariance, and we do not go further. The last three may be interpreted as follows: we modify a shape by some process and we compute the distance between the original shape and the new one. Then, it should be small in the following cases:

1. blurring: we add some parts, possibly important, but close to the shape.
2. occlusion: we hide a small part of a shape (possibly changing its topology).
3. noise addition: we add small parts possibly far from the shape.

#### Simplicity

This last property is not an axiom properly speaking, since it is not related to the recognition itself. However, we believe that an algorithm will be all the more efficient and fast, if it manipulates a small amount of data. Intuitively, it is certainly easier to describe the curve of Fig. 1.2(b) than the one of Fig. 1.2(a). For instance, we could think of keeping a sketch of the curve linking the points with maximal curvature. On the noisy curve, nearly all the points are maxima of curvature and the sketch is as complex as the original curve.

### 1.1.2 ... and their consequences

What can be deduced from the heuristic above? First, in order to get insensitivity to noise, it seems natural to smooth the shapes. Naturally, we then face the problem: what kind of oscillation can be labelled as noise, or contains real information? There is no absolute answer to this question but it shall only depend of a single parameter called *scale* representing the typical size of what will be considered as noise, or the distance at which we observe the shape. Since we cannot choose this scale a priori, smoothing will be multiscale and shape recognition will have a sense at each scale. Since the recognition must resist to occlusion, it should, at least partially, rely on local features. This is another argument for smoothing since local features are sensitive to noise. For instance, commonly used are the inflexion points and the maxima of curvature. Since they are defined from second derivatives, this is clear that a noisy curve as in Fig. 1.2(a) is totally unreliable.

## 1.2 Curve smoothing

We now admit the principle that shape recognition is made possible by a multiscale smoothing process removing the noise at each scale. It seems, that by adding this step, we have complicated the problem. Indeed. We do not know what kind of smoothing we have to choose, the only objective we have is to make local features reliable. It is also obvious that the smoothing has to be compatible with all the assumptions we made on the recognition task (invariance, stability, simplicity). The