An Elastica-driven Digital Curve Evolution Model for Image Segmentation

Daniel Antunes · Jacques-Olivier Lachaud · Hugues Talbot

Abstract Geometric priors have been shown to be useful in image segmentation to regularize results. For example, the classical Mumford-Shah functional uses region perimeter as prior. This has inspired much research in the last few decades, with classical approaches like the Rudin-Osher-Fatemi and most graph-cut formulations, which all use a weighted or binary perimeter prior. It has been observed that this prior is not suitable in many applications, for example for segmenting thin objects or some textures, which may have high perimeter/surface ratio. Mumford observed that an interesting prior for natural objects is the Euler Elastical model, which involves the squared curvature. In other areas of science, researchers have noticed that some physical binarization processes, like emulsion unmixing can be well-approximated by curvature-related flow like the Willmore flow. However, curvature-related flows are not easy to compute because curvature is difficult to estimate accurately, and the underlying optimisation processes are not convex. In this article, we propose to formulate a digital flow that approximates an Elastica-related flow using a multigrid-convergent curvature estimator, within a discrete variational framework. We also present an application of this model as a post-processing step to a segmentation framework.

Keywords Multigrid convergence · Digital estimator · Curvature · Shape Optimization · Image Segmentation.

Daniel Antunes
Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000 Chambéry, France
E-mail: daniel.martins-antunes@univ-smb.fr

Jacques-Olivier Lachaud
Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000 Chambéry, France
E-mail: jacques-olivier.lachaud@univ-smb.fr

Hugues Talbot
CentraleSupelec, Inria, Université Paris-Saclay, 9 rue Joliot-Curie, F-91190 Gif-sur-Yvette
E-mail: hugues.talbot@centralesupelec.fr

1 Introduction

Geometric quantities are particularly useful as regularizers in low-level image analysis, especially when little prior information is known about the shape of interest. Length penalization is a well-behaved, general purpose regularizer and many models in the literature make use of it, from active contours [?] to level-set formulations [26, 27]. Discrete graph-based variational models have been particularly succesfull to incorporate length penalization as a penalizer, while keeping the ability to extract a global optimum [5, 2].

However length regularization shows limitations when segmenting small or thin and elongated objects, as it tends to shrink solutions or yields disconnected solutions. Such drawbacks can be problematic in image segmentation or image inpainting. It is thus rather natural to consider curvature (especially squared curvature) as a potential regularizer. Energies involving both length and squared curvature are often called *Euler elastica* model (that dates back to Euler) or *Willmore energy* (*n*-dimensional extension) and have been brought back to attention in computer vision by Mumford [31]. A similar regularizer is also present in the second-order regularizer of the original snake model [20]. Similarly, an explicit curvature term was often employed in early level-set methods, like [26, 27, 3] for segmentation or inpainting, However, Casselles *et al.* in [?] observed that these terms were not geometric, in the sense that they depended on the discretization.

The use of curvature for surface regularization was studied in [4], although results look promising, authors used an inexact linearized version of the square curvature at every step.

One of the first successful uses of geometric squared curvature in image processing is the inpainting algorithm described in [30], where authors evaluate the elastica energy along the level lines of a simply-connected image to reconstruct its occluded parts. The non-intersection property of level lines induces the construction of an efficient dynamic programming algorithm. However, the curvature energy is only coarsely approximated. [9] is another geometric inpainting method involving elastica optimized with level-sets.

In image segmentation, Zehiry and Grady have shown that injecting curvature as a regularizer can help recover thin and elongated objects [15]. Similarly Schoenemann *et al.* [38] have considered the elastica energy in their ratio-based image segmentation approach. In [39], Schoenemann *et al.* extend ratio-cut approaches to segmentation with a curvature term. Their global optimization framework shows the considerable advantages of using such regularizer in common binary image segmentation tasks. However the time complexity of their algorithm is prohibitive, even if curvature is again coarsely approximated. In more recent work, Lim *et al.* propose an edge-weighted Willmore-like flow to segment vertebra [25] using a level-set formulation. They observe improved results compared to [?] and other perimeter-based priors, but observe that a shape prior and a close-enough solution must also be provided to achieve good results.

In fact, it is still a very challenging task to inject curvature in the context of image segmentation. State-of-art methods are in practice difficult to optimize and do not scale ======= known about the shape of interest. Length penalization is a well-behaved, general-purpose regularizer and many models in the literature make use of it, from active contours [8] to level-set formulations [26, 27]. Discrete graph-based variational models have been particularly successful to incorporate length penalization as a penalizer while keeping the ability to extract a global optimum [5, 2].

However, length regularization shows limitations when segmenting small or thin and elongated objects, as it tends to shrink solutions or yields disconnected solutions. Such drawbacks can be problematic in image segmentation, image restoration or image inpainting. It is thus rather natural to consider curvature (especially squared curvature) as a potential regularizer. Energies involving both length and squared curvature are often called the *Elastica* model (that dates back to Euler). The *Willmore energy* is its n-dimensional extension. These were brought to attention in computer vision by Mumford [31]. A similar concept is also present in the second-order regularizer of the original snake model [20]. Indeed, an explicit curvature term was often employed in early level-set methods, like [26, 27, 3] for segmentation or inpainting. However, Caselles *et al.* in [8] observed that these terms as employed were not geometric, in the sense that they depend on the discretization parameters.

1.1 Existing works

The use of curvature for surface regularization in the case of the Willmore energy was studied in [4]. Although results look promising, the authors used an inexact linearized version of the square curvature at every step.

One of the first successful uses of curvature in image processing is the inpainting algorithm described in [30], where authors evaluate the absolute curvature along the level lines of a simply-connected image to reconstruct its occluded parts. The non-intersection property of level lines induces the construction of an efficient dynamic programming algorithm. The curvature energy is nevertheless only coarsely approximated. We may quote [9] as another geometric inpainting method involving Elastica, which is optimized with level-sets.

In image segmentation, Zehiry and Grady have shown that injecting curvature as a regularizer can help recover thin and elongated objects [15]. Similarly, Schoenemann *et al.* [38] have considered the Elastica energy in their ratio-based image segmentation approach. In [39], Schoenemann *et al.* extend ratio-cut approaches to segmentation with a curvature term. Their global optimization framework shows the considerable advantages of using such regularizer in common binary image segmentation tasks. However, the time complexity of their algorithm is prohibitive, even if curvature is again coarsely approximated. In a more recent work, Lim *et al.* propose an edge-weighted Willmore-like flow to segment vertebra [25] using a level-set formulation. They observe improved

results compared to [8] and other perimeter-based priors, but observe that a shape prior and a close-enough initialization must also be provided to achieve good results.

In fact, it is still a very challenging task to efficiently handle curvature in the context of image segmentation. State-of-art methods are in practice difficult to optimize and do not scale easily [15, 38, 40, 32]. In order to achieve reasonable running times, such approaches make use of coarse curvature estimations for which the approximation error is often unknown. In the segmentation method of [38], for example, the curvature is estimated by a simple formula involving angles and edge lengths at polygon vertices. The quality of the estimation is higly influenced by the number of vertices and where they are located along the curve. More important, in this estimator, the curve representation lies in the continuous domain and the properties of such estimator are not immediately valid when the curve is translated to the digital domain. In particular, it is not straightforward to include an image data term in this model without adding extra assumptions about the curve representation. One could use a spline approximation with exact curvature information and the same kind of approach would have the same pros and cons. Improving the quality of the curvature estimator has an important impact on the accuracy of the results, but we should use the appropriate tools to evaluate its quality. In the context of measurements in digital data, this tool is multigrid convergence.

Segmentation methods using Elastica or Willmore energies can be formulated via the computation of the corresponding flow, i.e. following its gradient. Well-founded level-set methods involving Willmore energy [14] have been used in medical image segmentation [25]. Another level-set formulation similar to the Chan-Vese model was proposed in [42]. Willmore flow can also be carried out by phase-field models (see [7]) but they are less suited to image segmentation since interfaces are blurred in such models. Threshold dynamics can also be considered for such energies [18] and has been proposed for image disocclusion [17]. However, flow-based methods will always deliver a local minimum, which heavily depend on initialization for good results. A similar approach is to formulate the Elastica as a regularization in a variational framework and use operator splitting to optimize it. Tai *et al.* [41] and more recently Deng *et al.* [13] have proposed this approach, which is well-suited to inverse problem solving. The problem remains non-convex and the nature of the converged iterates is not studied, i.e. how close they are to a global optimum.

In contrast, researchers have sought convex relaxation of curvature-related formulation. In [19], Goldluecke and Cremers introduce the *Total Curvature*, based on the Menger-Melnikov curvature of a measure. The formulation is non-convex but can be relaxed into a well-correlated but non-tight convex one. The relaxed formulation can be used to solve inverse problems including segmentation. However computing a solution is still expensive, requiring significant parallelisation efforts with a GPU to achieve acceptable computing times. In [6], Bredies *et al.* study a convex approximation of the Elastica energy, using a lifting scheme. Thanks to the lifting, it is not restricted to binary sets. It shows promising results and can be used for image restoration in ad-

dition to inpainting and segmentation. The scheme is, however, neither tight nor exact, and required 4 dimensions for a 2D approximation. In addition, it does not carry over to nD. It also remains computationally expensive.

Discrete approaches have been used to tackle the Elastica difficulties. As noted before, [38] have linearized a mesh-based approach, which was expanded upon by [40], while [15] have used a binary solver for a non-submodular approach on a regular square grid of pixels, which was expanded in [16]. In [32], a specific solver is developed to yield a better solution to an extension of the formulation in [15].

1.2 Motivation

In spite of the multiplicity of existing approaches, it seems none is currently completely satisfactory. While the sought solution is expected to be very smooth based on geometrical expectations, the mathematical properties of the related operators make this smoothness elusive. Non-convexity and the presence of high-order terms make Elastica difficult to optimize and to approximate, let alone compute efficiently. The computed solutions in the literature are rarely evaluated, and the complexity of the proposed solution is often high.

Consequently, we are interested in studying purely discrete variational segmentation models involving an Elastica energy. Recently, new estimators of curvature along digital contours have been proposed [36,11,37] with the desirable multigrid convergence property. This motivates us to propose models in which they can be used successfully.

In this paper, we propose to investigate the use of the digital integral invariant curvature estimator [11] in a discrete variational segmentation framework. More precisely, we show how to incorporate it in a digital flow minimizing its squared curvature. This model is expressible as a discrete combinatorial optimization model, which we solve using a classical variant of QPBO algorithm [35]. Our solution does not compute a global solution to the whole domain but a global solution to a band surrounding a given shape. Our approach thus leads to a digital flow similar to continuous Elastica, as shown by our experiments. We present an application of this model as a post-processing step in a segmentation framework and demonstrate how it can improve standard results. This paper is an extended version of a previous work that appeared in [1]. We have included a deeper discussion on our optimization method and we present a more thorough comparison of our segmentation method with two other related works. Moreover, we have included a new local combinatorial optimization approach for Elastica minimization, which shows that our discrete model is well-founded. Finally, the code related to this paper is freely available on GitHub.[1]

---

[1] https://www.github.com/danoan/BTools

1.3 Outline

Section 2 reviews the concept of multigrid convergence and highlights its importance for the definition of digital estimators. We describe two convergent estimators used in this paper, one that approaches the length of elementary discrete contour elements, and the other that approaches the curvature of a discrete contour. They are used in the optimization model and in the definition of the digital Elastica. Section 3 describes a local combinatorial optimization model suitable for several curvature estimators, that presents interesting results but is too costly in practice. Section 4 describes the proposed Elastica-driven curvature evolution model along with several illustrations of digital flows. Section 5 explains how to use this evolution model as a post-processing step in an image segmentation framework and compares it to two related approaches. We conclude and draw some perspectives to this work in Section 6.

2 Multigrid convergent estimators

Our objective is to delineate shapes within digital images with some priors related to continuous geometry. The goal of this section is to introduce the concept of multigrid convergence and its potential when analyzing digital images with variational models involving geometric quantities, with the constraint that only digitized shapes are observed.

A digital shape is the result of some quantization process over an object X lying in some continuous space of dimension 2 (here). For example, the Gauss digitization of a continuous subset X of the Euclidean plane $\mathbb{R}^2$ with grid step $h > 0$ is defined as

$$D_h(X) = X \cap (h\mathbb{Z})^2.$$

Given a shape X and its digitization $D_h(X)$, a digital estimator $\hat{u}$ for some geometric quantity u is intended to compute $u(X)$ by using only the digitization. This problem is not well-posed, as the same digital object could be the digitization of infinitely many objects very different from X. Therefore, a characterization of what constitutes a good estimator is necessary.

Let u be some geometric quantity of X (e.g. tangent, curvature). We wish to devise a digital estimator $\hat{u}$ for u. It is reasonable to state that $\hat{u}$ is a good estimator if $\hat{u}(D_h(X))$ converges to $u(X)$ as we refine our grid. For example, counting pixels is a convergent estimator for area (with a rescale of h^2); but counting boundary pixels (with a rescale of h) is not a convergent estimator for perimeter. Multigrid convergence is the mathematical tool that makes this definition precise. Given any subset Z of $(h\mathbb{Z})^2$, we can represent it as a union of axis-aligned squares with edge length h centered on the point of Z. The topological boundary of this union of cubes is called h-frontier of Z. When $Z = D_h(X)$, we call it h-boundary of X and denote it by $\partial_h X$.

Definition 1 (Multigrid convergence for local geometric quantites)
A local discrete geometric estimator $\hat{u}$ of some geometric quantity u is (uniformly) multigrid convergent for the family $\mathbb{X}$ if and only if, for any $X \in \mathbb{X}$, there exists a grid step $h_X > 0$ such that the estimate $\hat{u}(D_h(X), p, h)$ is defined for all $p \in \partial_h X$ with $0 < h < h_X$, and for any $x \in \partial X$,

$$\forall p \in \partial_h X \text{ with } \|p - x\|_\infty \leq h, \|\hat{u}(D_h(X), p, h) - u(X, x)\| \leq \tau_X(h),$$

where $\tau_X : \mathbb{R}^+ \setminus \{0\} \to \mathbb{R}^+$ has null limit at 0. This function defines the speed of convergence of $\hat{u}$ towards u for X.

For a global geometric quantity (e.g. perimeter, area, volume), the definition remains the same, except that the mapping between ∂X and $\partial_h X$ is no longer necessary.

Multigrid convergent estimators provide a quality guarantee and should be preferred over non-multigrid convergent ones. In the next section, we describe two estimators that are important for our purpose.

2.1 Tangent and Perimeter Estimators

The literature presents several perimeter estimators that are multigrid convergent (see [10,12] for a review), but in order to define the digital Elastica we need a local estimation of length and we wish that integration over these local length elements gives a multigrid convergent estimator for the perimeter.

Definition 2 (Elementary Length) Let a digital curve C be represented as a sequence of grid vertices in a grid cell representation of digital objects (in a grid with step h). Further, let $\hat{\mathbf{v}}$ be a multigrid convergent estimator for the unit tangent vector. The elementary length $\hat{s}(\mathbf{e})$ at some oriented grid edge $\mathbf{e} \in C$ is defined as

$$\hat{s}(\mathbf{e}) = h \; \hat{\mathbf{v}}(\mathbf{e}) \cdot \mathbf{e}.$$

The integration of the elementary length along the digital curve is a multigrid convergent estimator for perimeter if one uses the λ-MST [24] tangent estimator (see [23]).

2.2 Integral Invariant Curvature Estimator

Generally speaking, an invariant is a function whose value is unaffected by the action of some group on the elements of the domain.

Perimeter and curvature are examples of invariants for shapes on $\mathbb{R}^2$ with respect to the Euclidean group of rigid transformations. Definition of integral area invariant and its link with curvature is proven in [28].

Definition 3 (Integral area invariant) Let $X \subset \mathbb{R}^2$ and $B_r(p)$ the ball of radius r centered at point p. Further, let $\mathbb{1}_X(\cdot)$ be the characteristic function of X. The integral area invariant $\sigma_{X,r}(\cdot)$ is defined as

$$\forall p \in \partial X, \quad \sigma_{X,r}(p) = \int_{B_r(p)} \mathbb{1}_X(x)dx.$$

The value $\sigma_{X,r}(p)$ is the area of the intersection of the ball $B_r(p)$ with shape X. By approaching the shape at point $p \in X$, one can rewrite the intersection area $\sigma_{X,r}(p)$ in the form of the Taylor expansion [33]:

$$\sigma_{X,r}(p) = \frac{\pi}{2}r^2 - \frac{\kappa(X,p)}{3}r^3 + O(r^4),$$

where $\kappa(X,p)$ is the curvature of X at point p. By isolating κ we can define a curvature estimator

$$\tilde{\kappa}(p) := \frac{3}{r^3}\left(\frac{\pi r^2}{2} - \sigma_{X,r}(p)\right), \tag{1}$$

Such an approximation is convenient as one can simply devise a multigrid convergent estimator for the area.

Definition 4 Given a digital shape $D \subset (h\mathbb{Z})^2$, the area estimator $\widehat{\mathrm{Area}}(D, h)$ is defined as

$$\widehat{\mathrm{Area}}(D, h) := h^2 \mathrm{Card}\left(D\right). \tag{2}$$

It is well known that this area estimator is multigrid convergent (e.g. see [21]). In [11], the authors combine the approximation(1) and digital estimator (2) to define a multigrid convergent estimator for the curvature.

Definition 5 (Integral Invariant Curvature Estimator) Let $D \subset (h\mathbb{Z})^2$ a digital shape. The integral invariant curvature estimator is defined as

$$\hat{\kappa}_r(D, p, h) := \frac{3}{r^3}\left(\frac{\pi r^2}{2} - \widehat{\mathrm{Area}}\left(B_r(p) \cap D, h\right)\right).$$

This estimator is multigrid convergent with speed $O(h^{\frac{1}{3}})$ for radii chosen as $r = \Theta(h^{\frac{1}{3}})$. This estimator is also robust to noise and can be extended to estimate the mean curvature of three dimensional shapes.

2.3 Elastica energy estimator

In the remaining of this article, we are considering the minimization of the Elastica energy. Given a Euclidean shape X with smooth enough boundary, the Elastica is defined as

$$E(X) = \int_{\partial X} (\alpha + \beta \kappa^2) ds, \quad \text{for } \alpha \geq 0, \beta \geq 0. \tag{3}$$

The digital version of Elastica energy approaching the continuous Elastica energy is defined as follows (it uses multigrid convergent estimators):

$$\hat{E}(D_h(X)) = \sum_{\mathbf{e} \in \partial D_h(X)} \hat{s}(\mathbf{e}) \left(\alpha + \beta \hat{\kappa}_r^2(D_h(X), \dot{\mathbf{e}}, h) \right), \tag{4}$$

where $\dot{\mathbf{e}}$ denotes the center of the edge $\mathbf{e}$. In the expression above, we will substitute an arbitrary subset S of $\mathbb{Z}^2$ to $D_h(X)$ since the continuous shape X is unknown. In the following, we omit the grid step h to simplify expressions (or, putting it differently, we assume that the shape of interest is rescaled by $1/h$ and we set $h = 1$).

3 Local combinatorial optimization

The objectives of this section is to show both the potential of minimizing a purely digital Elastica energy but also to underline the difficulties of using it in a global combinatorial optimization framework.

Given a digital shape $S^{(0)}$ we describe a process that generates a sequence $S^{(i)}$ of shapes with non-increasing Elastica energy. The idea is to define a neighborhood of shapes $\mathcal{N}^{(i)}$ to the shape $S^{(i)}$ and choose the element of $\mathcal{N}^{(i)}$ with lowest energy. The process is suited for the integral invariant estimator but also for other curvature estimators, for example, MDCA [36]. As a matter of fact, our experiments have shown that either estimator induce similar results.

Let S be a 2-dimensional digital shape embedded in a domain $\Omega \subset \mathbb{Z}^2$. We adopt the cellular-grid model to represent S, i.e., pixels and its lower dimensional counterparts, linels and pointels, are part of S (see figure 1). In particular, we denote by ∂S the topological boundary of S, i.e., the connected sequence of linels such that for each linel we have one of its incident pixels in S and the other not in S.

Let $d_S : \Omega \to \mathbb{R}$ be the signed Euclidean distance transformation with respect to shape S. The value $d_S(p)$ gives the Euclidean distance between $p \notin S$ and the closest pixel in S. For points $p \in S$, $d_S(p)$ gives the distance between p and the closest pixel not in S.

Definition 6 (m-Ring Set) Given a digital shape $S \in \Omega$, its signed distance transformation d_S and natural number $m \neq 0$, the *m-ring set of S* is defined as
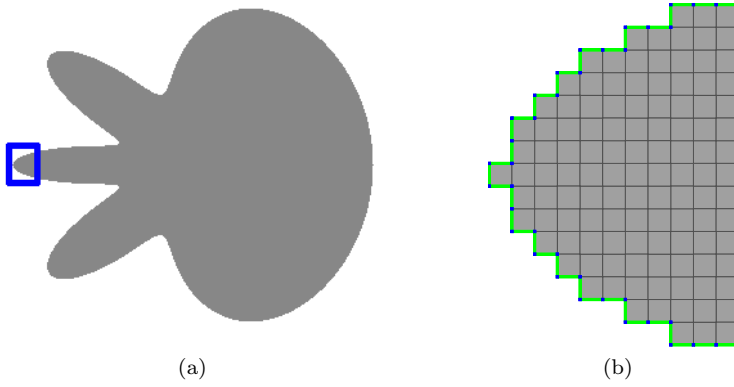
$$R_m(S) := L_m \cup L_{-m},$$

Fig. 1: The flower shape in figure (a) and the cellular-grid model representation in (b) of the rectangle-bounded region. In figure (b), pixels are colored in gray, linels in green and pointels in blue.

where

$$L_m(S) := \begin{cases} \{p \in \Omega \mid m - 1 < d_S(p) \leq m\}, & m > 0 \\ \{p \in \Omega \mid m + 1 > d_S(p) \geq m\}, & m < 0 \end{cases}$$

Consider the following set of neighbor candidates to S:

$$\mathcal{U}(S) = \{D \mid D \subset R_1(S) \cup S \text{ and } D \text{ is connected}\}.$$

Such set can be extremely large and its complete exhaustion is prohibitively expensive. Instead, we are going to use a subset of it.

Definition 7 (n-neighborhood) Given a digital shape $S \in \Omega$, its n-neighborhood $\mathcal{N}_n(S)$ is defined as the set of digital shapes that can be built from S by adding or removing a sequence of $k \in [0, n]$ connected pixels in $R_1(S)$.

Algorithm 1 describes the local combinatorial process and Figure 3 presents the digital curve evolution when executing this algorithm for two different shapes with $n = 50$.

The running time of algorithm 1 is summarized in table 1. All the experiments in this paper were executed on a 32-core $2.4Ghz$ CPU and the number of pixels in the square and flower shapes are respectively $841, 1641$ for grid step $h = 1.0$. Although its use in practical applications is limited, we demonstrate that digital estimators are effective in their measurements and the flows evolve as expected. In particular, the digital elastica energy for the experiments in figure 3 approaches the global optimum (figure 4). We observe that it is a complete digital approach, and we do not suffer from discretization and rounding problems, a common issue in continuous models. Furthermore, we have checked that this approach works indifferently with Integral Invariant
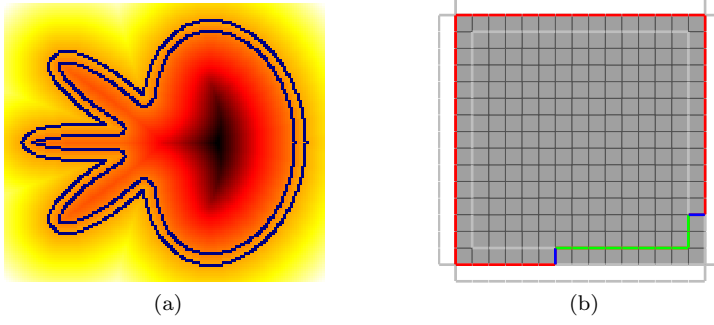
(a)   (b)

Fig. 2: The blue pixels in figure (a) illustrates a 3-ring set. In figure (b) we highlight the contour of an element of set $\mathcal{N}_{11}$.

---

input : A digital set S; the neighborhood parameter n; the maximum number of iterations maxIt; and a stop condition tolerance

1 delta $\longleftarrow$ tolerance $+1$;
2 $i \longleftarrow 0$;
3 **while** $i <$ maxIt **and** delta $>$ tolerance **do**
4   **for** $X \in \mathcal{N}_n(S^{(i)})$ **do**
5   **if** $\hat{E}(X) < \hat{E}(X^\star)$ **then**
6              $X^\star \longleftarrow X$
7   **end**
8   **end**
9      $i \longleftarrow i + 1$;
10     $S^{(i)} \longleftarrow X^\star$;
11   delta $\longleftarrow \hat{E}(S^{(i-1)}) - \hat{E}(S^{(i)})$;
12 **end**

Algorithm 1: Local combinatorial optimization for Elastica minimization.

---

curvature estimator and Maximal Digital Circular Arc curvature estimator. So the convergence of the digital curvature estimator seems to be the cornerstone to get a digital curve behaving like a continuous Elastica. In the next section, we explore a more efficient approach.

4 Digital Curvature Flow

Given a digital shape $S \subset \Omega \subset \mathbb{Z}^2$, the goal is to compute a flow $S^{(i)}$ where $S^{(i+1)}$ has lower digital elastica than $S^{(i)}$.

We observe that the integral invariant estimator (1) is originally designed to be evaluated on the digital boundary of the shape, which is the unknown of our problem. Injecting contour information leads to a third-order model, which we prefer to avoid. Instead, we are going to exploit the II estimator definition to compute a flow that evolves proportionally to the squared curvature of the current shape.
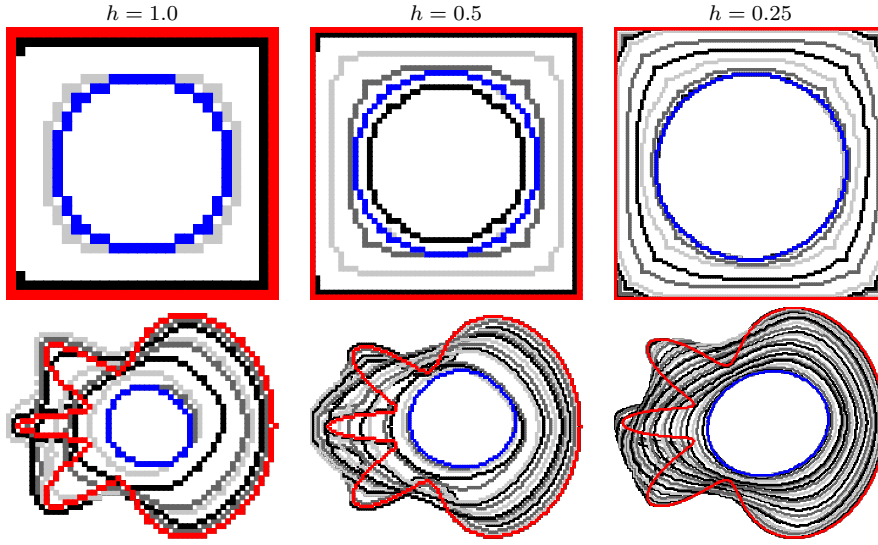
Fig. 3: Local combinatorial optimization process results using II estimator with radius 5 for the square and flower shapes. Shapes are displayed at every 5 iterations.

	Square	Flower
$h = 1.0$	4s	124s
$h = 0.5$	77s	1924s
$h = 0.25$	1886s	28445s

Table 1: Running times for the local combinatorial optimization algorithm with $n = 50$. Thirty two threads were used.

4.1 Discrete variational model for discrete curve evolution

We assume an ordering in Ω, i.e., there exists a bijective function $\omega : \Omega \to \{1 \cdots |\Omega|\}$. Moreover, we associate to any subset $P \subset \Omega$ the set of binary variables $X(P)$ defined as

$$X(P) := \left\{ x_{\omega(p)} \in \{0,1\} \mid p \in P \right\}.$$

In order to guarantee connectivity, we limit the optimization region to a subset of Ω, namely the inner pixel boundary of $S^{(i)}$. We also use the notation $|S|$ to denote the cardinal of a digital set S.

Definition 8 (Inner pixel boundary)
Given a digital shape S embedded in a domain Ω, we define its inner pixel boundary set $I(S)$ as

$$I(S) := \left\{ p \mid p \in S, |\mathcal{N}_4(x) \cap S| < 4 \right\},$$
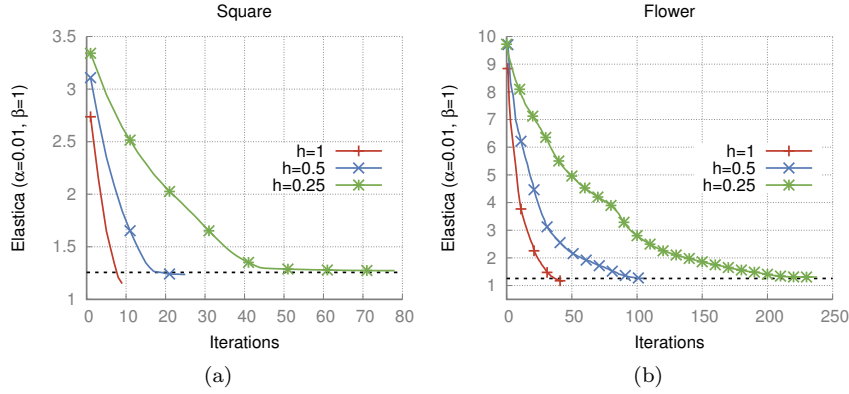
Fig. 4: Digital elastica computation along iterations of algorithm 1. The final shape has energy value close to the optimal value $2\pi/5$ (thin horizontal dashed line).

where $\mathcal{N}_4(p)$ denotes the 4-adjacent neighbor set of p (without p).

To simplify notation, the inner pixel boundary of $S^{(i)}$ is simply denoted $I^{(i)}$. At each iteration, the set $X^{(i)}$ of optimization variables is defined as

$$X^{(i)} := X(I^{(i)}).$$

In other words, each flow iteration decides which pixels in the inner boundary are to be removed from $S^{(i)}$ and which are to be kept in $S^{(i)}$. We recall the definition of the II digital curvature estimator:

$$\hat{\kappa}^2(p) = c_1 \left(c_2 - |B_r(p) \cap S^{(i)}| \right)^2, \tag{5}$$

where $c_1 = 3/r^6$ and $c_2 = \pi r^2/2$.

The following sets are important in the expansion of (5).

$$F^{(i)} := S^{(i)} \setminus I^{(i)}$$
$$F_r^{(i)}(p) := F^{(i)} \cap B_r(p)$$
$$I_r^{(i)}(p) := I^{(i)} \cap B_r(p)$$
$$X_r^{(i)}(p) := X\left(I_r^{(i)}(p)\right).$$
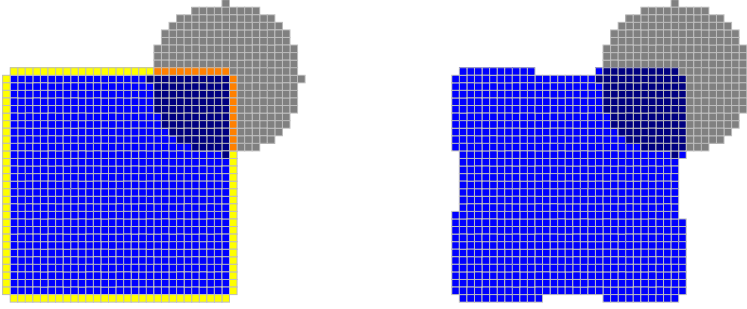
Expanding (5), we get

Fig. 5: Directly using the optimization result of (6) doesn't decrease squared curvature because contour information is not present in the energy.

$$\hat{\kappa}^2(p) = c_1 \Big(c_2 - |F_r^{(i)}(p)| - \sum_{x_j \in X_r^{(i)}(p)} x_j \Big)^2$$

$$= c_1 \Big(C + 2 \Big(|F_r^{(i)}(p)| - c_2 \Big) \sum_{x_j \in X_r^{(i)}(p)} x_j + \sum_{x_j \in X_r^{(i)}(p)} x_j^2 + \sum_{\substack{x_j, x_k \in X_r^{(i)}(p) \\ j < k}} 2 x_j x_k \Big),$$

where $C = c_2^2 - 2c_2 \cdot |F_r^{(i)}(p)| + |F_r^{(i)}(p)|^2$ is a constant. By ignoring constants and multiplication factors and using the binary character of the variables, we define the following family of energies for given parameters $\alpha, \beta, \gamma \geq 0$.

$$E_m(X^{(i)}, S^{(i)}) = \sum_{x \in X^{(i)}} \alpha s(x) +$$

$$\sum_{\substack{p \in \\ R_m(S^{(i)})}} 2c_1 \beta \Big((1/2 + |F_r^{(i)}(p)| - c_2) \cdot \sum_{\substack{x_j \in \\ X_r^{(i)}(p)}} x_j + \sum_{\substack{j < k, \\ x_j, x_k \in \\ X_r^{(i)}(p)}} x_j x_k \Big) +$$

$$\sum_{x \in X^{(i)}} \gamma g(S^{(i)}, x), \tag{6}$$

where $g(S, x)$ denotes a data term (e.g., distance to initial shape S or fidelity to data) and $s(x)$ denotes a length penalization term. Each choice of m generates a different flow, which is generally described in the Digital Curve Evolution (DCE) algorithm 2.

We emphasize that the minimization of (6) for $S^{(i)}$ is not sufficient to derive the next shape $S^{(i+1)}$ because contour information is not included in the model (see figure 5). Recall that the integral invariant estimator approaches curvature by computing the difference between half of the area of a chosen ball and the area of the intersection of this ball with the shape. In particular, regions of positive curvature have fewer pixels in their intersection set than on its complement w.r.t the estimation ball. This implies that variables in such

```
      input  : A digital set S; The ring number m; Length(α), curvature(β) and data(γ)
                coefficients; the maximum number of iterations maxIt;
 1   S^(0) ⟵ S;
 2   i ⟵ 1;
 3   while i < maxIt do
 4   │   //Expansion mode
 5   │   if i is odd then
 6   │   │   X^(i) ⟵ arg min_X E_m(X, S^(i-1));
 7   │   │   S^(i) ⟵ F_S^(i-1) + X^(i);
 8   │   end
 9   │   //Shrinking mode
10   │   else
11   │   │   X^(i) ⟵ arg min_X E_m(X, S^(i-1));
12   │   │   S^(i) ⟵ F_S^(i-1) + X^(i);
13   │   end
14   │   i ⟵ i +1;
15   end
```

Algorithm 2: Digital curvature evolution algorithm (DCE).

regions are labeled with 1, as the unbalance grows otherwise. We attenuate curvature if we shift the center of the estimation ball towards the interior of the shape, which means removing the 1-labeled pixels. That is why we take the complement of the optimization solution.

The same reasoning applies for non-convex parts. Indeed, concave regions are convex in the shape complement. In the expansion mode we apply the same reasoning on the image complement, and by doing this we are able to handle concavities. It is called expansion mode because the optimization region, in this case, is the outer pixel boundary of the original shape. Table 2 sums up these arguments.

Length and data terms should be properly defined in order to comply with the complement step of the DCE algorithm. The length penalization is defined as

$$s(x_{w(p)}) = \sum_{q \in \mathcal{N}_4(p)} t(q), \quad \text{where } t(q) = \begin{cases} (x_{w(p)} - x_{w(q)})^2, & \text{if } q \in I^{(i)} \\ (x_{w(p)} - 0), & \text{if } q \in F^{(i)} \\ (x_{w(p)} - 1), & \text{otherwise} \end{cases} \quad (7)$$

We do not use data terms in this section, thus we postpone the definition of such terms until later in this article, with the description of how DCE can be used in an image segmentation framework.

In figure 6 we see some results for the DCE algorithm with $m = 1$. We observe a global evolution towards rounder shapes, but with several artifacts. We minimize the effects of a jaggy boundary by setting $\alpha > 0$. Nonetheless, a higher radius of the estimation ball creates unstable shapes. In fact, the estimator is very sensitive in regions of low squared curvature, and it is precisely in those regions that spurious pixels are created.

shrink mode	$\kappa \gg 0$	$\kappa \geq 0$	$\kappa < 0$
X	$x_k = 1$	$x_k \in \{0,1\}$	$x_k = 0$
$S^{(i+1)} \leftarrow S^{(i)} \setminus X$	eroded	prob. eroded	unchanged
expansion mode	$\bar{\kappa} \gg 0$	$\bar{\kappa} \geq 0$	$\bar{\kappa} < 0$
$\bar{X}$	$\bar{x}_k = 1$	$\bar{x}_k \in \{0,1\}$	$\bar{x}_k = 0$
$S^{(i+1)} \leftarrow \overline{\overline{S^{(i)}} \setminus \bar{X}}$	dilated	prob. dilated	unchanged

Table 2: Since the curvature is negated when reversing the curve (i.e. $\bar{\kappa} = -\kappa$), this process can only shrink convex parts in shrink mode and expand concave parts in expansion mode.

(a) $r = 3, \alpha = 0$ (b) $r = 3, \alpha = 0.15$ (c) $r = 5, \alpha = 0.15$
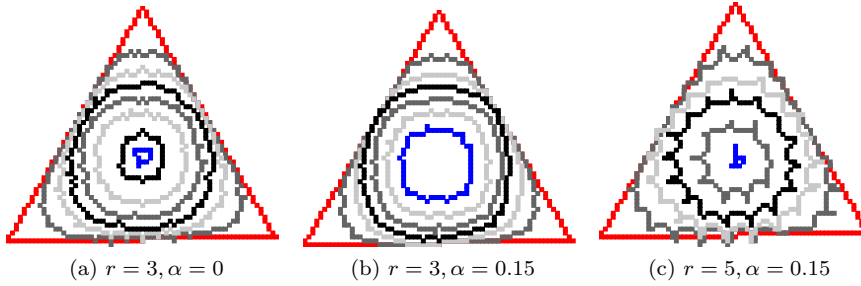
Fig. 6: The algorithm is very sensitive to the little variations of the estimator, which are particularly important in regions of low squared curvature. Artifacts are somewhat reduced with a length penalization but increases if we use a higher ball radius.

4.2 A more stable model.

In the previous section we noticed that the algorithm produces shapes with many artifacts due to the small uncertainties of the estimator along regions of low squared curvature. We argue that, by evaluating the estimation ball along outer ring sets we avoid those sensitive areas by focusing the optimization process only on regions with highest squared curvature value.

In our experiments, the best results are obtained by executing DCE algorithm with m equal to r, where r is the estimation ball radius (see figure 7). We observe that digital elastica may increase after some iterations if chosen radius is too large, as in the case of the triangle in figure 8 in which the flow converges to a single point. We conjecture that an appropriated value for the radius should be given by the shape reach. The produced flow has no difficulties in handling changes on topology, and it presents different speeds for regions with low and high curvature values, as illustrated in figure 9.
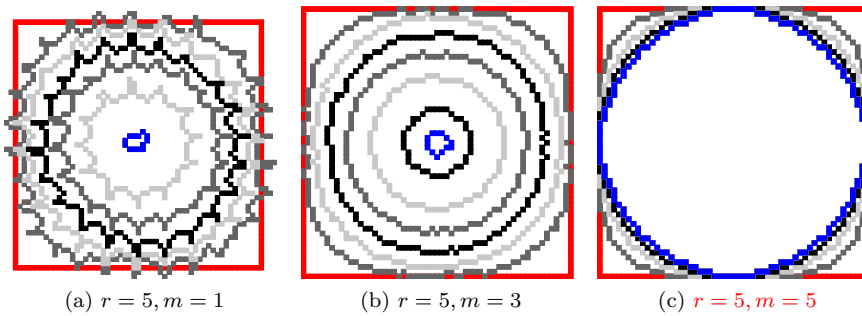
(a) $r = 5, m = 1$ (b) $r = 5, m = 3$ (c) $r = 5, m = 5$

Fig. 7: By positioning the estimation ball on farther rings, we minimize artifacts creation.

4.3 Optimization method

Let f be a function of n binary variables with unary and pairwise terms, i.e.

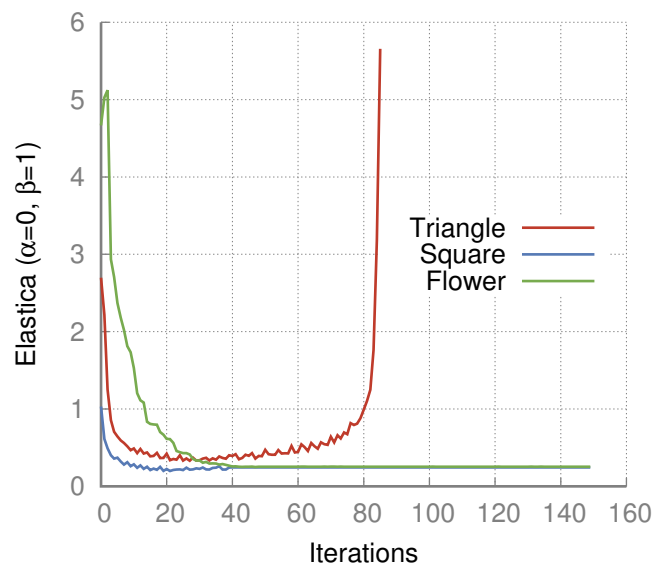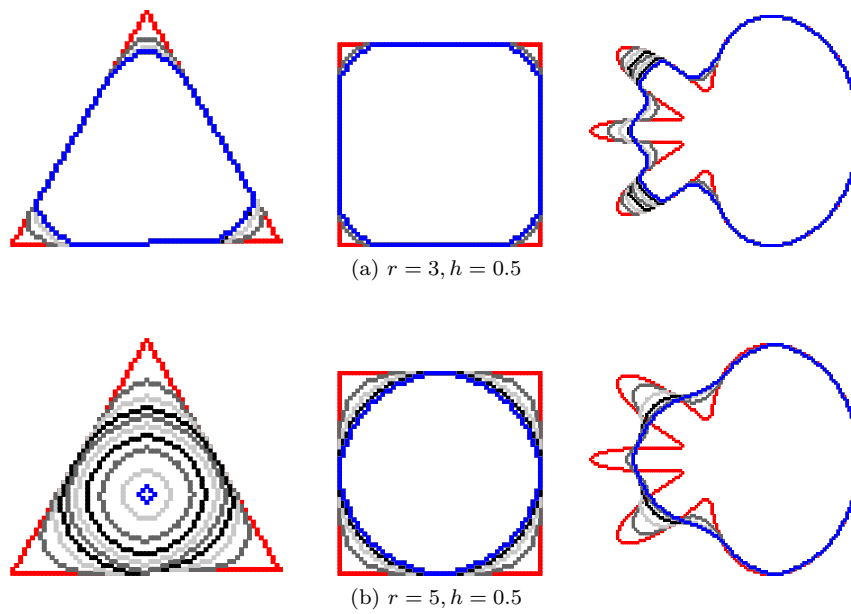$$f(y_1, \cdots, y_n) = \sum_j f_j(y_j) + \sum_{j<k} f_{j,k}(y_j, y_k).$$

The function f is submodular if and only if the following inequality holds for each pairwise term $f_{j,k}$ [22]:

$$f_{j,k}(0,0) + f_{j,k}(1,1) \le f_{j,k}(0,1) + f_{j,k}(1,0).$$

The energy E_m is non-submodular and optimizing it is a difficult problem, which constrains us to use heuristics and approximation algorithms. The QPBO method [35] transforms the original problem in a max-flow/min-cut formulation and yields a full optimal labeling for submodular energies. For non-submodular energies the method is guaranteed to return a partial labeling with the property that the set of labeled variables is part of an optimal solution. That property is called partial optimality.

In practice, QPBO can leave many pixels unlabeled. There exist two extensions to QPBO that alleviate this limitation: QPBOI (improve) and QPBOP (probe). The first is an approximation method that is guaranteed to not increase the energy, but loses the property of partial optimality. The second is an exact method which is reported to label more variables than QPBO.

The percentage of unlabeled pixels by QPBOP for E_1 is quite high, but the percentage decreases to zero as we set m equal to r. Therefore, we are more confident in taking the solution for values of m close to r. However, the way it varies across values of m differs from shape to shape, as is illustrated in figure 11. We also noticed that, for $m = r$, all the pixels were labeled, which may indicate that E_r is an easy instance of the general non-submodular energy E_m, but this remains to be proved. The number of pairwise terms in E_r is roughly half of those in E_1 (see figure 10).

(a) $r = 3, h = 0.5$

(b) $r = 5, h = 0.5$

(c) Digital elastica evaluation $(r = 5, h = 0.5)$

Fig. 8: The choice of radius impacts the flow. In the figures, the flow ceases to evolve for all shapes when $r = 3$ (a). In figure (b), for $r = 5$, the triangle evolves to a single point, while the others stop in an intermediate shape, as in (a). In figure (c), we observe that for a given choice of radius, the digital elastica may increase after a certain number of iterations.
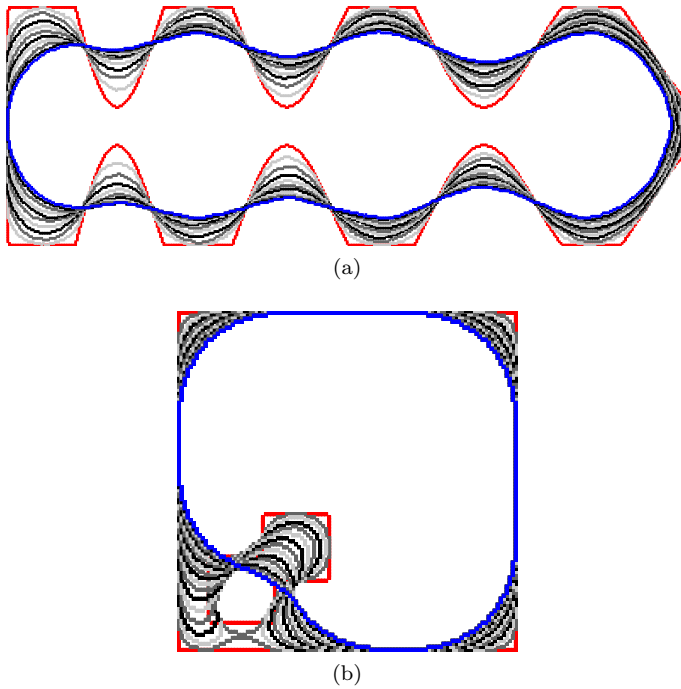
(a)

(b)

Fig. 9: High curvature regions evolves faster than lower ones (a). The flow can handle topological changes (b).

We have used QPBOI to solve E_m. Naturally, in the case where all pixels are labeled by QPBOP, QPBOI returns the same labeling as QPBOP.

5 Application to image segmentation

We present an application of our digital curve evolution algorithm to supervised image segmentation. The DCE acts as a contour correction method. Here we use a data fidelity term in order to characterize the object of interest. Given foreground and background seeds selected by the user, we derive mixed Gaussian distributions of color intensities G_f, G_b, and we define the data fidelity term as the cross-entropy, i.e.

$$g(x_{w(p)}) = -(1 - x_{w(p)}) \log G_f(p) - x_{w(p)} \log G_b(p). \qquad (8)$$

We use the DCE algorithm to regularize an initial contour output by some segmentation algorithm or delineated by the user. In this application, the data term of the DCE is set to the data fidelity term (8).

The algorithm can be initialized by a collection of compact sets, or with the result of a third-party segmentation algorithm, as GrabCut [34]. We include
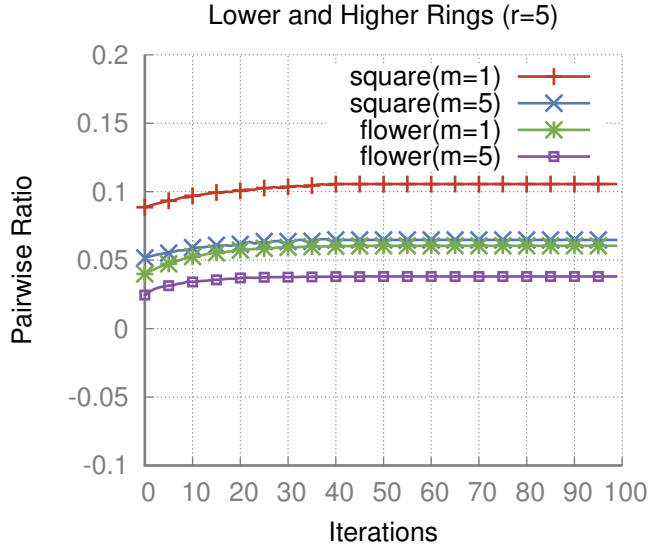
Fig. 10: We plot the ratio of pairwise terms among all $\binom{|X^{(i)}|}{2}$ combinations. The highest ring has roughly half the number of pairwise terms as the lowest ring.

---

input : An image I; seed mask M; the estimation ball radius r; length (α), squared curvature (β) and data fidelity (γ) coefficients; initial dilation d; stop condition value tolerance; the maximum number of iterations maxIt;

1 $S \longleftarrow \mathrm{GrabCut}(I, M)$;
2 $S^{(0)} \longleftarrow \mathrm{dilate}(S, d)$;
3 delta $\longleftarrow +\infty$;
4 $i \longleftarrow 0$;
5 **while** $i < $ maxIt **and** delta $>$ tolerance **do**
6 $\quad$ $S^{(i+1)} \longleftarrow \mathrm{DCE}(S^{(i)}, r, \alpha, \beta, \gamma, 2)$;
7 $\quad$ delta $\longleftarrow |S^{(i)} - S^{(i+1)}|$;
8 $\quad$ i $\longleftarrow$ i +1;
9 **end**

Algorithm 3: Contour correction algorithm.

---

an additional parameter d that dilates the initial sets using a square of side one before executing the flow.

We evaluate our method using the BSD300 database [29]. All images contain the same number of pixels, the resolution being 321x481 (481x321) in portrait (landscape) mode. We compare the results of our method with segmentations given by GrabCut and Schoenemanns's method [38]. We report an average of $3s$ per flow iteration, and an average of 30 iterations per image. While GrabCut executes in less than one second, Schoenemann's method may take several hours to complete.
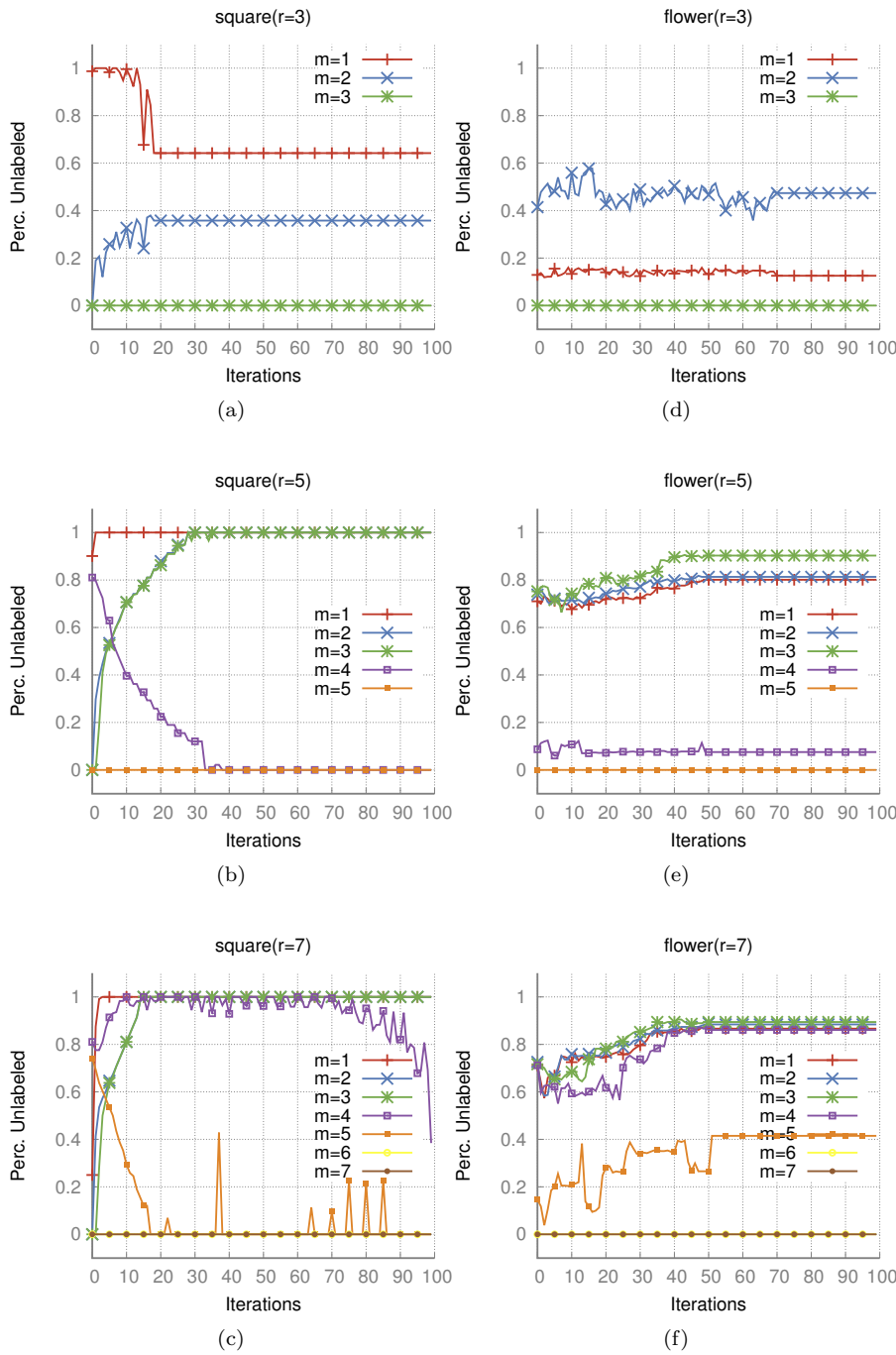
Fig. 11: For each plot, we first produce shapes $\{S^{(i)}\}$ executing DCE with $m = r$. Then, for each shape in $\{S^{(i)}\}$, we execute one iteration of DCE for different values of m and we count the unlabeled pixels. The number of unlabeled pixels by QPBOP remains high for lower values of m, and goes to zero when $m = r$. We observe the same behaviour for varying radius values.
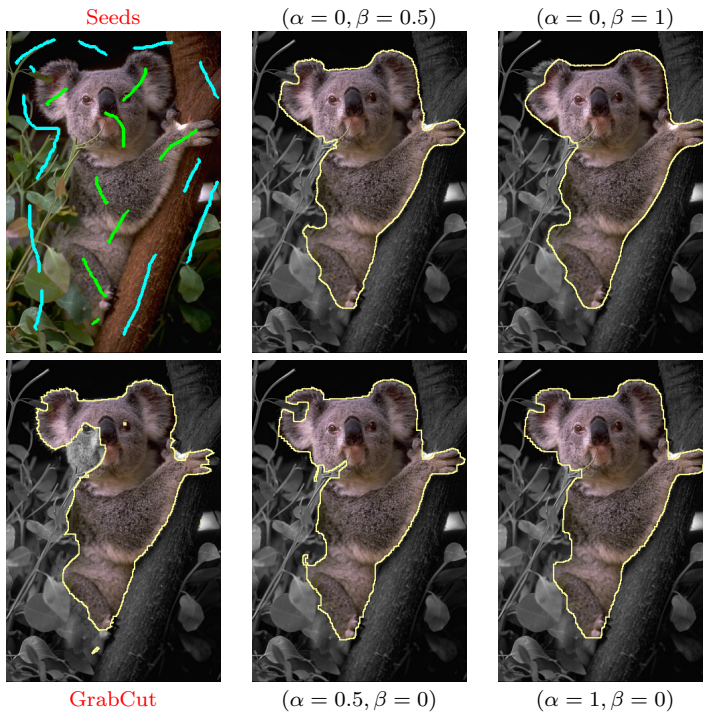
Fig. 12: Comparison of squared curvature regularization (first row) and length regularization (second row).

In figure 12 we can observe the results of a curvature regularization in comparison with a pure length regularization. The curvature can fill gaps and is smoother than the one produced by length only, resulting in more pleasant segmentations. In figures 13 and 14 we list several results of our method and we compare them with segmentation produced by GrabCut and Schoenemann's method. Our method is successful in producing curvature regularized segmentations and demonstrates the completion property of curvature. Moreover, it does not suffer from over-segmentation, it is much faster than Schoenemann's, and in several cases produces better segmentations than GrabCut.

6 Conclusion

We have studied in depth several digital curve evolution models based on a digital version of the Elastica energy and we have presented an application to image segmentation. The processes we have described are completely digital and do not suffer from issues that typically arise in models that pass through a discretization stage, such as rounding. Moreover, the model can handle changes in topology and its results are competitive with similar approaches while achieving reasonable running times.

GrabCut   Schoenemann   Contour Correction
                      $(\alpha = 0.1, \beta = 1.0)$ $(r = 5, \alpha = 0.1, \beta = 1.0, \gamma = 3.0)$
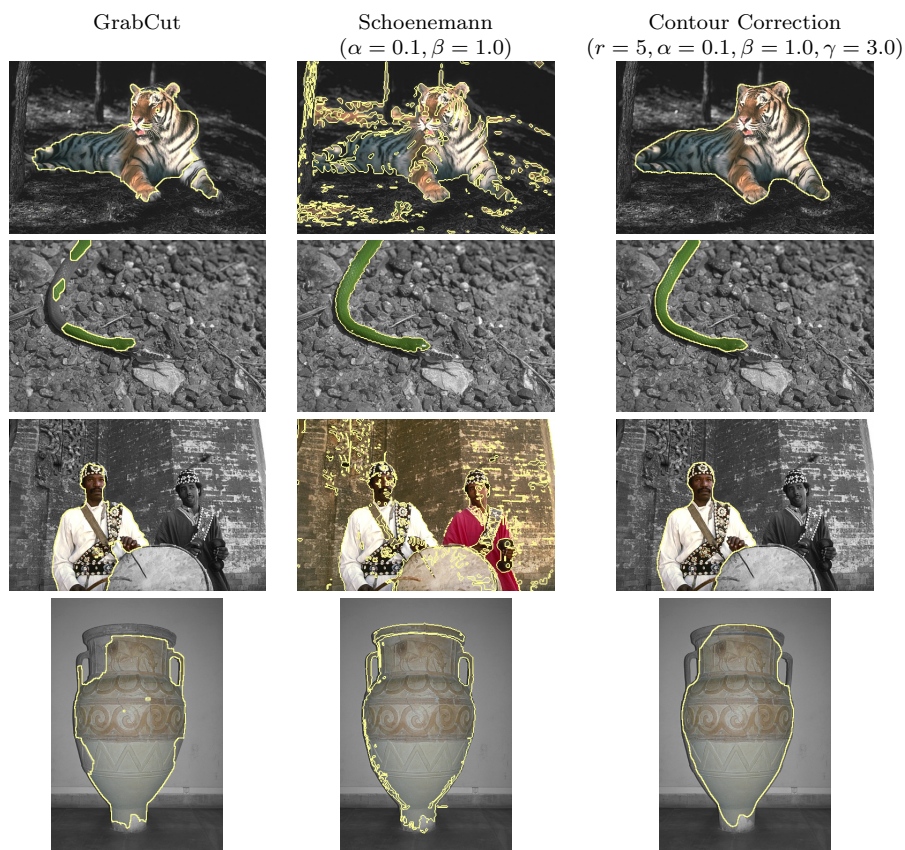

Fig. 13: The proposed method regularizes GrabCut [34] contours and returns meaningful results. We can observe the completion feature of curvature in the second row, and we don't suffer from over-segmentation issues as Schoenemann's method [38]. However, our flow may stop in a local optimum as in the fourth row, while Schoenemann's is able to extrapolate such solutions.

Future developments of this work will include extending the DCE algorithm to 3d, as the integral invariant estimator is available in higher dimensions. As a result, a denoising application may be derived by lifting the flow to 3d with a data fidelity defined by image intensities or by distance to initial position.

References

1. Antunes, D., Lachaud, J.O., Talbot, H.: Digital curvature evolution model for image segmentation. In: Couprie, M., Cousty, J., Kenmochi, Y., Mustafa, N. (eds.) Discrete Geometry for Computer Imagery. pp. 15–26. Springer International Publishing, Cham (2019)

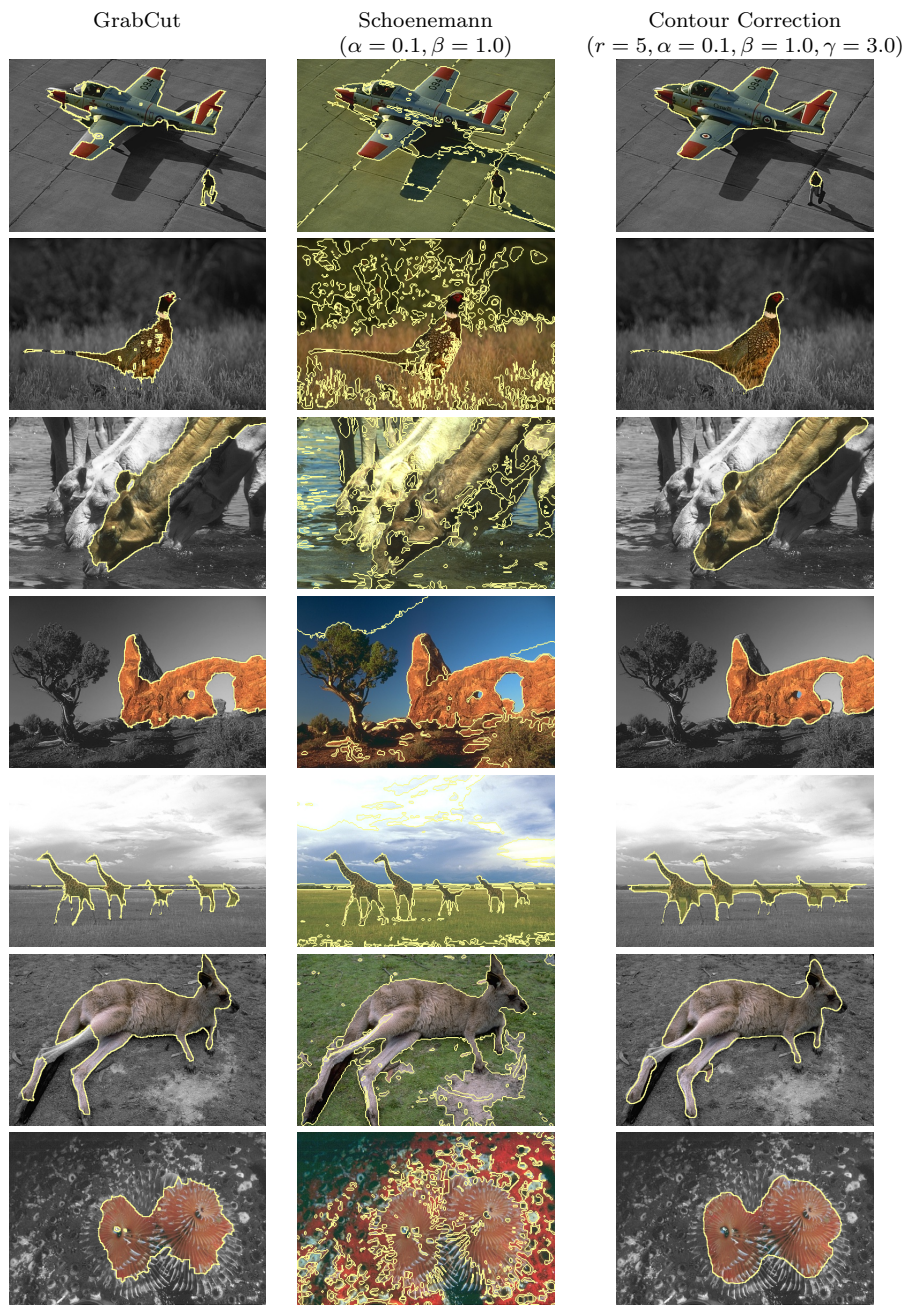| GrabCut	Schoenemann $(\alpha = 0.1, \beta = 1.0)$	Contour Correction $(r = 5, \alpha = 0.1, \beta = 1.0, \gamma = 3.0)$

Fig. 14: Additional comparison results between GrabCut [34], Schoenemman [38] and Contour correction segmentation methods.

2. Appleton, B., Talbot, H.: Globally optimal geodesic active contours. Journal of Mathematical Imaging and Vision **23**(1), 67–86 (Jul 2005)
3. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Transactions on Image Processing **10**(8), 1200–1211 (Aug 2001). https://doi.org/10.1109/83.935036
4. Bobenko, A.I., Schröder, P.: Discrete willmore flow (2005)
5. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001. vol. 1, pp. 105–112 (2001)
6. Bredies, K., Pock, T., Wirth, B.: A convex, lower semicontinuous approximation of Euler's elastica energy. SIAM Journal on Mathematical Analysis **47**, 566–613 (01 2015). https://doi.org/10.1137/130939493
7. Bretin, E., Masnou, S., Oudet, E.: Phase-field approximations of the willmore functional and flow. Numerische Mathematik **131**(1), 115–171 (2015)
8. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. International Journal of Computer Vision **22**(1), 61–79 (Feb 1997)
9. Chan, T.F., Kang, S.H., Kang, Shen, J.: Euler's elastica and curvature based inpaintings. SIAM J. Appl. Math **63**, 564–592 (2002)
10. Coeurjolly, D., Klette, R.: A comparative evaluation of length estimators of digital curves. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(2), 252–258 (Feb 2004)
11. Coeurjolly, D., Lachaud, J.O., Levallois, J.: Integral based curvature estimators in digital geometry. In: Gonzalez-Diaz, R., Jimenez, M.J., Medrano, B. (eds.) Discrete Geometry for Computer Imagery. pp. 215–227. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
12. Coeurjolly, D., Lachaud, J.O., Roussillon, T.: Multigrid Convergence of Discrete Geometric Estimators, pp. 395–424. Springer Netherlands, Dordrecht (2012)
13. Deng, L.J., Glowinski, R., Tai, X.C.: A new operator splitting method for the Euler elastica model for image smoothing. SIAM Journal on Imaging Sciences **12**(2), 1190–1230 (2019)
14. Droske, M., Rumpf, M.: A level set formulation for willmore flow. Interfaces and Free Boundaries **6**(3), 361–378 (2004)
15. El-Zehiry, N.Y., Grady, L.: Fast global optimization of curvature. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 3257–3264 (June 2010)
16. El-Zehiry, N.Y., Grady, L.: Contrast driven Elastica for image segmentation. IEEE Transactions on Image Processing **25**(6), 2508–2518 (2016)
17. Esedoglu, S., Ruuth, S., Tsai, R.: Threshold dynamics for shape reconstruction and disocclusion. In: IEEE International Conference on Image Processing 2005. vol. 2, pp. II–502. IEEE (2005)
18. Esedoglu, S., Ruuth, S.J., Tsai, R.: Threshold dynamics for high order geometric motions. Interfaces and Free Boundaries **10**(3), 263–282 (2008)
19. Goldluecke, B., Cremers, D.: Introducing total curvature for image processing. In: 2011 International Conference on Computer Vision. pp. 1267–1274 (Nov 2011). https://doi.org/10.1109/ICCV.2011.6126378
20. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision **1**(4), 321–331 (1988)
21. Klette, R., Žunić, J.: Multigrid convergence of calculated features in image analysis. Journal of Mathematical Imaging and Vision **13**(3), 173–191 (2000)
22. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(2), 147–159 (Feb 2004)
23. Lachaud, J.O.: Non-Euclidean spaces and image analysis : Riemannian and discrete deformable models, discrete topology and geometry. Ph.D. thesis, Université Sciences et Technologies - Bordeaux I (Dec 2006), https://tel.archives-ouvertes.fr/tel-00396332
24. Lachaud, J.O., Vialard, A., de Vieilleville, F.: Fast, accurate and convergent tangent estimation on digital contours. Image Vision Comput. **25**(10), 1572–1587 (Oct 2007)

25. Lim, P.H., Bagci, U., Bai, L.: Introducing willmore flow into level set segmentation of spinal vertebrae. IEEE Transactions on Biomedical Engineering **60**(1), 115–122 (2012)
26. Malladi, R., Sethian, J.A.: Image processing via level set curvature flow. Proceedings of the National Academy of sciences **92**(15), 7046–7050 (1995)
27. Malladi, R., Sethian, J.A., Vemuri, B.C.: Shape modeling with front propagation: A level set approach. IEEE T. Patt. Anal. Mach. Intell. **17**(2) (1995)
28. Manay, S., Hong, B.W., Yezzi, A.J., Soatto, S.: Integral invariant signatures. In: Pajdla, T., Matas, J. (eds.) Computer Vision - ECCV 2004. pp. 87–99. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
29. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. 8th Int'l Conf. Computer Vision. vol. 2, pp. 416–423 (July 2001)
30. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269). pp. 259–263 vol.3 (Oct 1998)
31. Mumford, D.: Elastica and computer vision. In: Algebraic geometry and its applications, pp. 491–506. Springer (1994)
32. Nieuwenhuis, C., Toeppe, E., Gorelick, L., Veksler, O., Boykov, Y.: Efficient squared curvature. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 4098–4105 (June 2014)
33. Pottmann, H., Wallner, J., Huang, Q.X., Yang, Y.L.: Integral invariants for robust geometry processing. Computer Aided Geometric Design **26**(1), 37 – 60 (2009)
34. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": Interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. **23**(3), 309–314 (Aug 2004)
35. Rother, C., Kolmogorov, V., Lempitsky, V.S., Szummer, M.: Optimizing binary MRFs via extended roof duality. 2007 IEEE Conference on Computer Vision and Pattern Recognition pp. 1–8 (2007)
36. Roussillon, T., Lachaud, J.O.: Accurate curvature estimation along digital contours with maximal digital circular arcs. In: Aggarwal, J.K., Barneva, R.P., Brimkov, V.E., Koroutchev, K.N., Korutcheva, E.R. (eds.) Combinatorial Image Analysis. pp. 43–55. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
37. Schindele, A., Massopust, P., Forster, B.: Multigrid convergence for the MDCA curvature estimator. J. Math. Imaging Vis. **57**(3), 423–438 (Mar 2017)
38. Schoenemann, T., Kahl, F., Cremers, D.: Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 17–23 (Sept 2009)
39. Schoenemann, T., Masnou, S., Cremers, D.: The elastic ratio: Introducing curvature into ratio-based image segmentation. IEEE Transactions on Image Processing **20**(9), 2565–2581 (2011)
40. Strandmark, P., Kahl, F.: Curvature regularization for curves and surfaces in a global optimization framework. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) Energy Minimization Methods in Computer Vision and Pattern Recognition. pp. 205–218. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
41. Tai, X.C., Hahn, J., Chung, G.J.: A fast algorithm for Euler's elastica model using augmented lagrangian method. SIAM J. Img. Sci. **4**(1), 313–344 (Mar 2011)
42. Zhu, W., Tai, X.C., Chan, T.: Image segmentation using Euler's elastica as the regularization. Journal of Scientific Computing **57**(2), 414–438 (2013)