Resolution Independent Deformable Model

Blinded for Submission
Blinded for Submission
E-mail: Blinded for Submission

Abstract

In this paper we propose a parametric deformable model that automatically adapts its topology and that recovers accurately image components with a complexity independent from the resolution of input image. The main idea is to equip the image space with a metric that expands interesting features in the image depending on their geometry.

1 Introduction

In the field of image analysis, recovering image components is a difficult task. This turns out to be even more challenging when objects exhibit large variations of their shape and topology. Deformable models that are able to handle that kind of situation can use only little *a priori* knowledge concerning image components. This generally imply heavy computational costs.

In the framework of parametric deformable models, most authors [5, 11, 12] propose to investigate the intersections of the deformable model with a grid that covers the image space. Special configurations of these intersections characterize the self collisions of the mesh. Local reconfigurations are then performed to adapt the topology of the model according to its geometry. To take advantage of all image details, the resolution of the grid should be the same as that of the image. An other method [9] consists in constraining the lengths of the edges of the model between two bounds. Self-collisions are then detected when distances between non-neighbor vertices fall under a given threshold. Topological consistency is recovered using local operators that reconnect vertices consistently. Using all image details requires edges to have the same size as image pixels. The complexities of all these methods are thus directly determined by the size of input data.

In the framework of level-set methods, boundaries of objects are implicitly represented as the zero level set of a function f [2, 3, 10, 16]. Usually f is sampled over a regular grid that has the same resolution as the input image. Then f is iteratively updated to make its zero level-set approach image contours. Even with optimization methods which reduce computations to a narrow band around evolving boundaries [1, 14], the complexity of these methods is determined by the resolution of the grid and hence by the resolution of the input image.

In [6] a method is proposed to adapt the resolution of a deformable model depending on its position and orientation in the image. The main idea is to equip the image space with a Riemannian metric that geometrically expands parts of the image with interesting features. Then, the length of edges is kept as uniform as possible with this new metric. In this first attempt the metric was manually given by a user. Our contribution is to propose an automated way of building metrics directly from images. Accuracy of the reconstruction is determined by the geometry of recovered image components: vertices accumulate in highly curved parts of the images and get sparser elsewhere. By this way the number of vertices on the mesh is optimized and the quality of shape representation is maintained. Properly resampling input images does not affect the geometry of image components, and the number of vertices is left invariant. Furthermore, vertices are allowed to travel faster in places with no image structure. The number of iterations and hence the time required to reach convergence are thus reduced.

2 Deformable Model

2.1 General description

Our proposed deformable model follows the classical energy formulation of active contours [8]: it is the discretization of a curve that is emdedded in the image space. Each of its vertices undergoes forces that regularize the shape of the curve, attract them toward image features and possibly tailor the model behavior [4, 15] for more specific purposes. In this paper, classi-

cal parametric snakes are extended with the ability to (i) dynamically and automatically change their topology in accordance with their geometry and (ii) adapt their resolution to take account of the geometrical complexity of recovered image components.

2.2 Resolution adaptation

During the evolution of the model the vertex density along the curve is kept as regular as possible by maintaining length constraints on edges:

$$\delta \leq L_E(u,v) \leq \zeta\delta \ , \tag{1}$$

where L_E denotes the length of the line segment that joins u and v. Parameter δ determines lengths of edges and hence vertex density along the curve. Parameter ζ determines the allowed ratio between maximum and minimum edge lengths.

At every step of the evolution of the model, each edge is checked. If its length is found to be less than δ then it is contracted. In contrast, if its length exceeds the $\zeta\delta$ threshold then it gets split.

Adaptive resolution is achieved by replacing the Euclidean length estimator L_E by a position-dependent length estimator L_R in (1). In places where L_R underestimates distances, estimated lengths of edges tend to fall under the δ threshold. As a consequence, edges tend to contract and the resolution of the model locally decreases. In contrast, the resolution of the model increases in regions where L_R overestimates distances.

More formally, Riemannian geometry provides us with theoretical tools to build such a distance estimator. In this framework, the length of an elementary displacement $\mathbf{ds}$ that starts from point (x,y) is expressed as:

$$\|\mathbf{ds}\|_R^2 = {}^t\mathbf{ds} \times G(x,y) \times \mathbf{ds}, \tag{2}$$

where G associates a positive-definite symmetrical matrix with each point of the space. The G mapping is called a *Riemannian metric*. From (2) follow the definitions of the Riemannian length of a path as

$$L_R(\gamma) = \int_a^b \|\dot{\gamma}(t)\| \ dt \ , \tag{3}$$

and of the Riemannian distance between two points u and v as

$$d_R(u,v) = \inf_{\gamma\in\mathcal{C}} L_R(\gamma) \ , \tag{4}$$

where $\mathcal{C}$ contains all the paths that join u and v. It is thus easily seen that defining the G mapping is enough to completely define our new length estimator L_R. How this mapping is built from images to enhance and speed up shape recovery is discussed in Sect. 3.

2.3 Topology adaptation

During the evolution of the model, care must be taken to ensure that its interior and exterior are always well defined: self-collisions are detected and the topology of the model is updated accordingly.

Since all edges have their length lower than $\zeta\delta$, a vertex that crosses over an edge (u,v) must approach either u or v closer than $\frac{1}{2}(\zeta\delta + d_{max})$, where d_{max} is the largest distance covered by a vertex during one iteration. Self-intersections are thus detected by looking for pairs of non-neighbor vertices (u,v) for which

$$d_E(u,v) \leq \frac{1}{2}(\zeta\delta + d_{max}) \ . \tag{5}$$

It is easily shown that this detection algorithm remains valid when d_E is replaced with a d_R distance estimator as described in Sect. 2.2. With a naive implementation, the complexity of this method is quadratic. However, it may be reduced to $O(n \log n)$ by storing vertex positions in an appropriate quadtree structure.

Detected self-intersections are solved using local operators that restore a consistent topology of the mesh by properly reconnecting the parts of the curve involved in the collision.

2.4 Dynamics

Theoretically, in a space equipped with a Riemannian metric the position $\mathbf{x}$ of a vertex that undergoes a force $\mathbf{F}$ follows equation

$$m\ddot{x}_k = F_k - \sum_{i,j} \Gamma_{ij}^k \dot{x}_i \dot{x}_j \ , \tag{6}$$

where the Γ_{ij}^k coefficients are known as the *Christoffel's symbols* associated with the metric. However, the last term of (6) has no influence once the model has reached its rest position and is second order in $\dot{\mathbf{x}}$. Therefore it is neglected and we get back the classical Newton's laws of motions.

3 Tailoring Metrics to Images

3.1 Geometrical interpretation

For any location (x,y) in the image space, the metric $G(x,y)$ is a positive-definite symmetrical matrix. Thus, in a orthonormal (for the Euclidean norm) base $(\mathbf{v}_1,\mathbf{v}_2)$ of eigenvectors, $G(x,y)$ is diagonal with coefficients (μ_1,μ_2). Hence, the length of an elementary displacement $\mathbf{ds} = x_1\mathbf{v}_1 + x_2\mathbf{v}_2$ is expressed as $\|\mathbf{ds}\|_R^2 = \mu_1 x_1^2 + \mu_2 x_2^2$. This shows that changing the

Euclidean metric with a Riemannian metric locally expands or contracts the space along $\mathbf{v}_1$ and $\mathbf{v}_2$ with ratios $1/\sqrt{\mu_1}$ and $1/\sqrt{\mu_2}$. The next paragraph explains how these eigenvectors and eigenvalues are computed from input images to enhance shape representation and to achieve independence from image resolution.

3.2 Required properties

In a place with no contour, vertex density is kept as low as possible. This reduces the cost of an iteration. Furthermore, this allows larger displacements of vertices at each iteration. As a consequence, the number of iterations required for the model to converge is reduced too.

In the neighborhood of image structures the deformable curve either follows image contours or crosses over it. In the former case, vertex density increases with both the strength and curvature of the contour. In the latter case, more degrees of freedom are given to the deformable model to allow it to get aligned more easily with the contour. In this case, vertex density is an increasing function of the strength of the contour.

Of course it is not possible to let edge length fall to 0 or increase too much. Therefore, eigenvalues of the metric are constrained between 1 and μ_m, and the length of edges remains between $\delta/\sqrt{\mu_m}$ and $\zeta\delta$.

The previous statements entail the following choice for the metric:

$$\left\{ \begin{array}{lll} \mathbf{v}_1 = \mathbf{n} & \text{and} & \mu_1 = \phi(s) \\ \mathbf{v}_2 = \mathbf{n}^\perp & \text{and} & \mu_2 = \psi(\kappa, s) \end{array} \right. , \qquad (7)$$

where $\mathbf{n}$ is a normal to the contour, and s and κ respectively denote the strength and curvature of the contour. Functions ϕ and ψ are defined as:

$$\phi(s) = \left[\frac{s^2}{s_m^2}\, \mu_m \right]_{1,\, \mu_m} \qquad \psi(\kappa, s) = \left[\frac{\kappa^2}{\kappa_m^2}\, \phi(s) \right]_{1,\, \phi(s)} \qquad (8)$$

where $[\cdot]_{a,b}$ constrains the values of its argument between bounds a and b. The parameter s_m determines the minimum strength for which a contour is considered as reliable. By this way, when the deformable model crosses over a contour, the length of its edges progressively decreases from δ for a weak contour to $\delta/\sqrt{\mu_m}$ for a contour with a strength greater than s_m. For a given strength of the contour, the κ_m parameter determines the curvature for which maximal resolution is reached. When the deformable curve runs along a straight reliable contour, the length of its edges is δ. When the curvature of the contour increases, the length of edges progressively decreases to $\delta/\sqrt{\mu_m}$ for contours with a curvature greater than κ_m.

The way parameters s and κ are computed directly from images is discussed in the next paragraph.

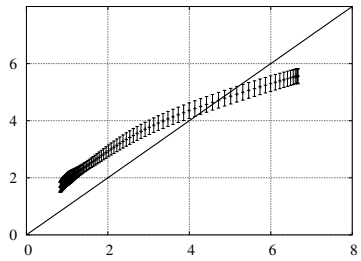

Figure 1. Estimated curvature (9) as a function of exact curvature. Solid line: ideal estimator.

3.3 Building metrics from images

The gradient structure tensor is a classical tool to investigate the coherence of image gradient over a neighborhood. It is defined as $J_{\rho,\sigma} = g_\rho * (\nabla I_\sigma \times {}^t\nabla I_\sigma)$, where g_ρ is the Gaussian filter with standard deviation ρ and I_σ denotes the input image smoothed with a Gaussian filter g_σ. At each point in the image, $J_{\rho,\sigma}$ is a positive-definite symmetrical matrix with eigendecomposition $\{(\mathbf{w}_1, \xi_1), (\mathbf{w}_2, \xi_2)\}$. Assuming that $\xi_2 \leq \xi_1$, the eigenvector $\mathbf{w_1}$ corresponds to the average direction of image gradient in a neighborhood, and $\mathbf{w}_2$ is tangent to image contours. Eigenvalues ξ_1 and ξ_2 correspond to the contributions of the gradient along these directions. They are used to assess the strength and curvature of contours of image components:

$$s^2 \simeq \xi_1 + \xi_2 \quad \text{and} \quad \kappa^2 \simeq \frac{1}{\rho^2} \times \frac{\xi_2}{\xi_1} \, . \qquad (9)$$

The estimator s is equivalent to the average norm of image gradient within a neighborhood. This curvature estimator κ is shown to be exact along the y-axis of the ideal images $I_\alpha(x, y) = y - \alpha x^2$. Moreover, it was tested on noisy images of ellipsoids with known curvature (see Fig 1). This approximation provides satisfying results. However, more elaborate estimators [13] could be used instead, but they are computationaly more expensive.

4 Experiments

Adaptive vertex density is illustrated in Fig. 2 (left). As expected, changing the metric increases vertex density along highly curved parts of image components, which is also visible in Fig. 3 (left and center).

Independence with respect to the resolution of input images is shown in Fig. 2 (right). Our model was tested on images of objects sampled at different rates (see Fig. 3). As expected, the number of vertices is kept independent from image resolution, as far the sampling

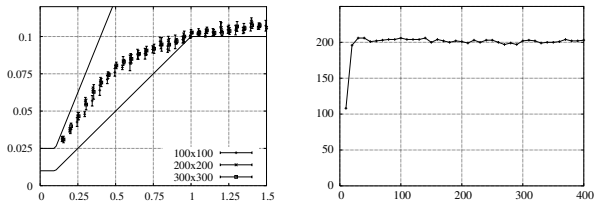

Figure 2. Left: Edge length as a function of the radius of curvature and for different resolutions of the input images. Solid lines correspond to the theoretical bounds. Right: final number of vertices on the curve depending on the resolution of input image.

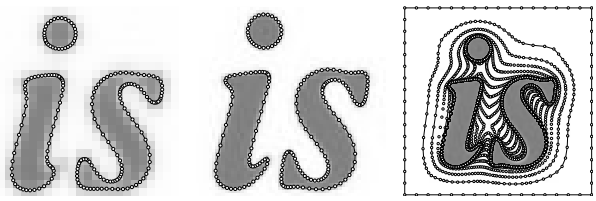

Figure 3. Left, center: reconstruction of identical objects sampled at resolutions 40×40 and 100×100. Right: evolution of the deformable model every 50 iterations.

rate ensures a proper representation of the highest frequencies in the input signal.

Fig. 3 demonstrates the ability of our model to dynamically change its topology. Moreover, it shows that the curve travels faster in parts of the image without feature and slows down and refines when approaching object boundaries. This is a consequence of expanding the space in the vicinity of image contours only.

5 Conclusion

We presented a deformable model that adapts its resolution according to the geometrical complexity of image features. It is therefore able to recover finest details in images with a complexity almost independent from the size of input data. All the material used in our presented deformable model has a straightforward extension to higher dimensions [7]. Further work will therefore extend our model for 3D image segmentation.

References

[1] D. Adalsteinsson and J. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.

[2] V. Caselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66, 1993.

[3] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proc. of ICCV95*, pages 694–699, Boston MA, June 1995.

[4] L. D. Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, mar 1991.

[5] H. Delingette and J. Montagnat. New algorithms for controlling active contours shape and topology. In D. Vernon, editor, *ECCV'2000*, number 1843 in LNCS, pages 381–395, Dublin, Ireland, June 2000. Springer.

[6] B. for submission. Blinded for submission. In *Blinded for submission*, Blinded for submission, 2002.

[7] B. for submission. Blinded for submission. In *Blinded for submission*. Blinded for submission, 2003.

[8] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.

[9] J.-O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine 3D surface extraction. *Medical Image Analysis*, 3(2):187–207, 1999.

[10] R. Malladi, J. A. Sethian, and B. C. Vemuri. Shape modelling with front propagation: A level set approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):158–174, feb 1995.

[11] T. McInerney and D. Terzopoulos. Medical image segmentation using topologically adaptable snakes. In *Proc. of Computer Vision, Virtual Reality and Robotics in Medicine*, pages 92–101, Nice, France, 1995. Springer.

[12] T. McInerney and D. Terzopoulos. Medical image segmentation using topologically adaptable surfaces. In J. Troccaz, E. Grimson, and R. Mösges, editors, *Proc. of CVRMed-MRCAS*, volume 1205 of *LNCS*, pages 23–32, Grenoble, France, mar 1997. Springer.

[13] B. Rieger and L. J. van Vliet. Curvature of n-dimensionnal space curves in grey-value images. *IEEE Trans. on Image Processing*, 11(7):738–745, July 2002.

[14] J. Strain. Tree methods for moving interfaces. *Journal of Computational Physics*, 15(2):616–648, May 1999.

[15] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. on Image Processing*, 7(3):359–369, mar 1998.

[16] A. Yezzi, Jr., S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum. A geometric snake model for segmentation of medical imagery. *IEEE Trans. on Medical Imaging*, 16(2):199–209, apr 1997.