

Geodesic Curvature Flow on Parametric Surfaces

Alon Spira and Ron Kimmel

Abstract. The motion of curves and images in \mathbb{R}^2 has been researched extensively. The curvature flow, which is a geometric heat equation for planar curves, is among the classical ones. In this paper we present a numerical scheme to extend the planar curvature flow to the geodesic curvature flow on parametric surfaces. The flow is implemented by back projecting the flow on the parametric surface to the parameterization plane, calculating the flow on the plane by the level sets method and then mapping it back to the manifold. This approach enables a more general and efficient solution of the problem than previous methods.

§1. Introduction

The motion of curves and images in \mathbb{R}^2 has been researched extensively. There are many applications in image processing and computer vision, such as scale space by linear and nonlinear diffusion, image enhancement through anisotropic diffusion, and image segmentation by snakes. The level sets formulation [9] has provided good means to implement these flows. Extending these motions to manifolds embedded in spaces of higher dimensions can be beneficial in computer graphics, computer vision, and image processing.

In this paper we demonstrate a numerical scheme to extend the curvature flow on the plane [4] to the geodesic curvature flow on parametric surfaces. The flow is implemented by back projecting the flow on the parametric surface to the parameterization plane, calculating the flow on the plane by the level sets method, and then mapping it back to the manifold. This technique has been used before for manifolds that are graphs of functions $(\{x, y, z(x, y)\})$ in order to find shortest paths [7,8], and to construct an intrinsic scale space for images on surfaces [6]. A similar approach was used for manifolds constructed from patches homeomorphic to \mathbb{R}^2 [2].

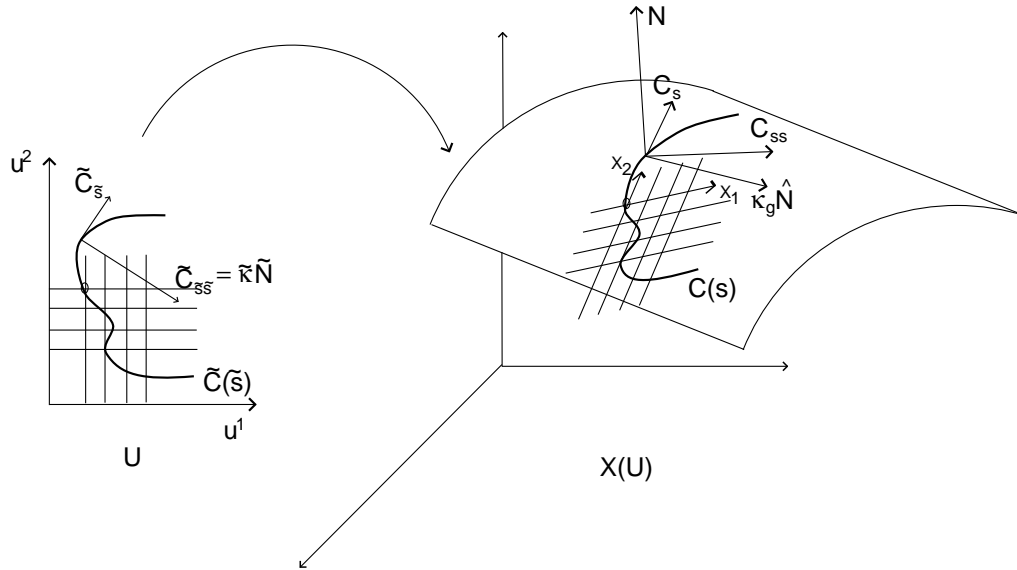


Fig. 1. The curve $C(s)$ on the manifold $X(U)$ and its origin $\tilde{C}(\tilde{s})$ on the parameterization plane U .

A different approach has been previously used for general manifolds [1]. It consisted of implicitly representing both the manifold and the curve on it as level sets of functions in \mathbb{R}^N (N - the dimension of the embedding space). This approach has several drawbacks, which are alleviated by our method. Unlike that method, our method does not require the extension of the manifold and the curve or data on it to functions in \mathbb{R}^N . The calculations are done implicitly on the parameterization plane and not in \mathbb{R}^N , which might be computationally prohibitive for $N > 3$. Finally, our method is not restricted to manifolds that can be represented by a level set, and we can thereby handle more general manifolds, such as self intersecting ones.

§2. Definitions and Motivation

We consider a parameterization plane $U = \{u^1, u^2\} \in \mathbb{R}^2$. This plane is mapped by $X : \mathbb{R}^2 \rightarrow \mathbb{R}^N$ to the parameterized surface $X(U) = \{x^1(u^1, u^2), x^2(u^1, u^2), \dots, x^N(u^1, u^2)\} \in \mathbb{R}^N$. Any curve $C(s) \in X(U)$ has an origin $\tilde{C}(\tilde{s}) \in U$, i.e., each point $p \in C(s)$ is a mapping of a corresponding point $\tilde{p} \in \tilde{C}(\tilde{s})$ by $p = X(\tilde{p})$. s and \tilde{s} are the arc length parameterizations of the curves C and \tilde{C} respectively. The derivatives of X with respect to u^i are defined as $X_i \equiv \frac{\partial X}{\partial u^i}$. See Figure 1.

According to the above definitions, the derivative of $C(s)$ with respect to its arc length is C_s , which is the tangent to the curve C . Similarly, we have $\tilde{C}_{\tilde{s}}$, which is the tangent to $\tilde{C}(\tilde{s})$. We denote by \mathcal{N} the normal to the plane tangent to the surface $X(U)$ and in the direction of $X_1 \times X_2$. $\hat{\mathcal{N}}$ is

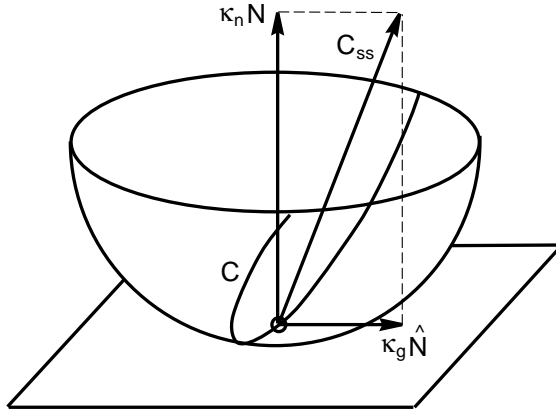


Fig. 2. The curvature has two orthogonal components: the normal curvature and the geodesic curvature.

the unit vector normal to the curve $C(s)$ lying in that plane. $\tilde{\mathcal{N}}$ represents the normal to \tilde{C}_s in the plane U .

Any geometric flow of the curve $C(s)$ of the form $C_t = F\hat{\mathcal{N}}$ has a corresponding geometric flow on U of the form $\tilde{C}_t = \tilde{F}\tilde{\mathcal{N}}$. If we can find \tilde{F} as a function of F and the mapping X , we can simplify the calculation of the flow on $X(U)$ by performing the flow on U , and then mapping the result onto $X(U)$. To enable this, we represent vectors in the N -dimensional space according to the basis $\{X_1, X_2\}$. The other components of the vectors, which are perpendicular to X_1 and X_2 , do not affect the flow of the curve $C(s)$ on the surface $X(U)$.

§3. Projecting the Flow on the Parameterization Plane

The geodesic curvature flow of $C(s)$ is

$$C_t = \kappa_g \hat{\mathcal{N}} = C_{ss} - \langle C_{ss}, \mathcal{N} \rangle \mathcal{N}. \quad (1)$$

This is the flow of the curve $C(s)$ according to the component of its curvature, tangent to the surface $X(U)$. Taking only this component of the curvature keeps the curve on the manifold. See Figure 2.

The representation of C_s according to the basis $\{X_1, X_2, \mathcal{N}\}$ is $C_s = u_s^i X_i$ (We use Einstein's summation convention in this expression and in the rest of the paper). By differentiating this expression with respect to s , we get

$$C_{ss} = u_{ss}^i X_i + u_s^i X_{ij} u_s^j = u_{ss}^i X_i + u_s^i (\Gamma_{ij}^k X_k + b_{ij} \mathcal{N}) u_s^j,$$

with Γ_{ij}^k being Christoffel's symbols and b_{ij} the coefficients of the second fundamental form [3]. $\kappa_g \hat{\mathcal{N}}$ is the component of C_{ss} in the plane tangent

to $X(U)$, i.e., the covariant derivative of C_s . We get it by discarding the component of C_s in the direction of \mathcal{N}

$$\kappa_g \hat{\mathcal{N}} = \frac{DC_s(s)}{ds} = u_{ss}^i X_i + \Gamma_{ij}^k X_k u_s^i u_s^j = (u_{ss}^k + \Gamma_{ij}^k u_s^i u_s^j) X_k. \quad (2)$$

If $C(s)$ is a geodesic, then by definition $\frac{DC_s(s)}{ds} = 0$, and (1) becomes $C_t = 0$. This is the stopping point of the geodesic curvature flow.

We use the chain rule to compute

$$C_t = \frac{\partial}{\partial t} X(\tilde{C}) = X_k u_t^k.$$

Combining this result with (1) and (2) yields

$$u_t^k = u_{ss}^k + \Gamma_{ij}^k u_s^i u_s^j,$$

or

$$\tilde{C}_t = \tilde{C}_{ss} + \{\Gamma_{ij}^1 u_s^i u_s^j, \Gamma_{ij}^2 u_s^i u_s^j\}. \quad (3)$$

In order to write this equation as a function of \tilde{s} instead of s , we use

$$\begin{aligned} s = \int |C_{\tilde{s}}| d\tilde{s} \quad \Rightarrow \quad q \equiv \frac{\partial \tilde{s}}{\partial s} &= |C_{\tilde{s}}|^{-1} = |X_i u_{\tilde{s}}^i|^{-1} \\ &= \left(X_i X_j u_{\tilde{s}}^i u_{\tilde{s}}^j \right)^{-\frac{1}{2}} = \left(g_{ij} u_{\tilde{s}}^i u_{\tilde{s}}^j \right)^{-\frac{1}{2}}, \end{aligned}$$

where we replaced $X_i X_j$ in q with g_{ij} , which are the components of the covariant metric tensor. We get

$$u_s^i = \frac{\partial \tilde{s}}{\partial s} u_{\tilde{s}}^i = q u_{\tilde{s}}^i$$

and

$$u_{ss}^i = q_s u_{\tilde{s}}^i + q u_{\tilde{s}s}^i = q_s u_{\tilde{s}}^i + q^2 u_{\tilde{s}\tilde{s}}^i.$$

Using these relations in (3) yields

$$\tilde{C}_t = q_s C_{\tilde{s}} + q^2 \left(C_{\tilde{s}\tilde{s}} + \{\Gamma_{ij}^1 u_{\tilde{s}}^i u_{\tilde{s}}^j, \Gamma_{ij}^2 u_{\tilde{s}}^i u_{\tilde{s}}^j\} \right),$$

but the geometric flow depends only on the component of \tilde{C}_t in the direction of $\tilde{\mathcal{N}}$, i.e.,

$$\langle \tilde{C}_t, \tilde{\mathcal{N}} \rangle = q^2 \left(\tilde{\kappa} + \langle \{\Gamma_{ij}^1 u_{\tilde{s}}^i u_{\tilde{s}}^j, \Gamma_{ij}^2 u_{\tilde{s}}^i u_{\tilde{s}}^j\}, \tilde{\mathcal{N}} \rangle \right) = \frac{\tilde{\kappa} + \langle \{\Gamma_{ij}^1 u_{\tilde{s}}^i u_{\tilde{s}}^j, \Gamma_{ij}^2 u_{\tilde{s}}^i u_{\tilde{s}}^j\}, \tilde{\mathcal{N}} \rangle}{g_{ij} u_{\tilde{s}}^i u_{\tilde{s}}^j}, \quad (4)$$

where $\tilde{\kappa}$ is the curvature of \tilde{C} .

We can compare this result with the result in [6] by defining $X = \{x, y, z(x, y)\}$. We calculate Γ_{ij}^k in this case by

$$\Gamma_{ij}^k = X_{ij} X^k,$$

where X^k is the contravariant version of the covariant vector X_k , calculated by

$$X^k = g^{kj} X_j,$$

where g^{ij} are the components of the contravariant metric tensor. This results in

$$\Gamma_{ij}^k = \frac{z_{ij} z_k}{g}, \quad (g = \det(g_{ij})).$$

Inserting this expression for Γ_{ij}^k in (4) yields

$$\langle \tilde{C}_t, \tilde{\mathcal{N}} \rangle = \frac{\tilde{\kappa} + \langle \frac{z_{ij}}{g} u_s^i u_s^j \{z_1, z_2\}, \tilde{\mathcal{N}} \rangle}{g_{ij} u_s^i u_s^j},$$

which can be easily shown to be equivalent to the expression in [6].

§4. Level Sets Representation of the Flow

We can convert the flow equation we got in the previous section into a level sets equation. This formulation enjoys many numerical advantages.

This means converting (4) into a level sets formulation. We assume that $\tilde{C}(\tilde{s}) = \{u^1(\tilde{s}), u^2(\tilde{s})\}$ is the zero set of $I(u^1, u^2)$. This means

$$I_t = \frac{\tilde{\kappa} + \langle \{\Gamma_{ij}^1 u_s^i u_s^j, \Gamma_{ij}^2 u_s^i u_s^j\}, \tilde{\mathcal{N}} \rangle}{g_{ij} u_s^i u_s^j} |\nabla I|. \quad (5)$$

To develop the expressions in the right hand side of (5) as functions of I , X , and their spatial derivatives, we use

$$\{-u_s^2, u_s^1\} = \tilde{\mathcal{N}} = \frac{\nabla I}{|\nabla I|} \quad \Rightarrow \quad u_s^1 = \frac{I_2}{(I_1^2 + I_2^2)^{\frac{1}{2}}}, \quad u_s^2 = \frac{-I_1}{(I_1^2 + I_2^2)^{\frac{1}{2}}} \quad (6)$$

and

$$\tilde{\kappa} = \operatorname{div} \left(\frac{\nabla I}{|\nabla I|} \right) = \frac{I_1^2 I_{22} - 2I_1 I_2 I_{12} + I_2^2 I_{11}}{(I_1^2 + I_2^2)^{\frac{3}{2}}}. \quad (7)$$

After some work we get

$$I_t = \frac{(-1)^{(i-j)} I_i I_j I_{(3-i)(3-j)}}{g |\nabla_M I|^2} + \frac{(-1)^{(i-j)} \Gamma_{ij}^k I_{(3-i)} I_{(3-j)} I_k}{g |\nabla_M I|^2} \quad (8)$$

with Christoffel's symbols calculated by derivatives of the first fundamental form

$$\Gamma_{ij}^k = \frac{1}{2}g^{kl}(\partial_i g_{lj} + \partial_j g_{il} - \partial_l g_{ij}).$$

If the surface is a plane, we expect that the geodesic curvature flow would become the curvature flow. In this case, $g_{11} = g_{22} = 1$, $g_{12} = 0$ and $\Gamma_{ij}^k = 0$, $\forall i, j, k$, and we get

$$I_t = \frac{I_1^2 I_{22} - 2I_1 I_2 I_{12} + I_2^2 I_{11}}{I_1^2 + I_2^2},$$

which is the level sets formulation of the curvature flow.

§5. The Numerical Scheme for the Level Sets Equation

An appropriate numerical scheme should be found for the implementation of (8). The first term on the right hand side of this equation is diffusive and can be implemented with central differences. The second term is a non-convex hyperbolic term and needs a special numerical scheme.

We used a fifth order Weighted Essentially Non-Oscillatory (WENO) scheme with a global Lax-Friedrichs (LF) flux in space [5] and a third order Total Variation Diminishing Runge-Kutta (TVD-RK) scheme in time. Non-periodic boundary conditions were used.

A re-distancing of the level sets function was activated every few iterations, as a regularizing process. The re-distancing was accomplished by the Sussman-Fatemi method [11]. This method uses the equation

$$\phi_t = \text{sign}(\phi_0)(1 - |\nabla\phi|)$$

to transform the level sets function ϕ_0 into a distance map. Also this equation is implemented by a fifth order WENO-LF, third order TVD-RK numerical scheme. The zero set of ϕ_0 is maintained by applying a volume conserving condition of the form

$$\partial_t \int_{\Omega} H(\phi) = 0,$$

with H the Heaviside function and Ω a fixed domain. The condition is applied by using a gradient projection step.

The fast marching algorithm [10] was also tested as a re-distancing method, but it lacks the necessary accuracy. This is due to the first order approximation method used to reconstruct the zero set of the level set function. The Sussman-Fatemi method, as opposed to fast marching, does not need to reconstruct the zero set, and can maintain its high order of accuracy.

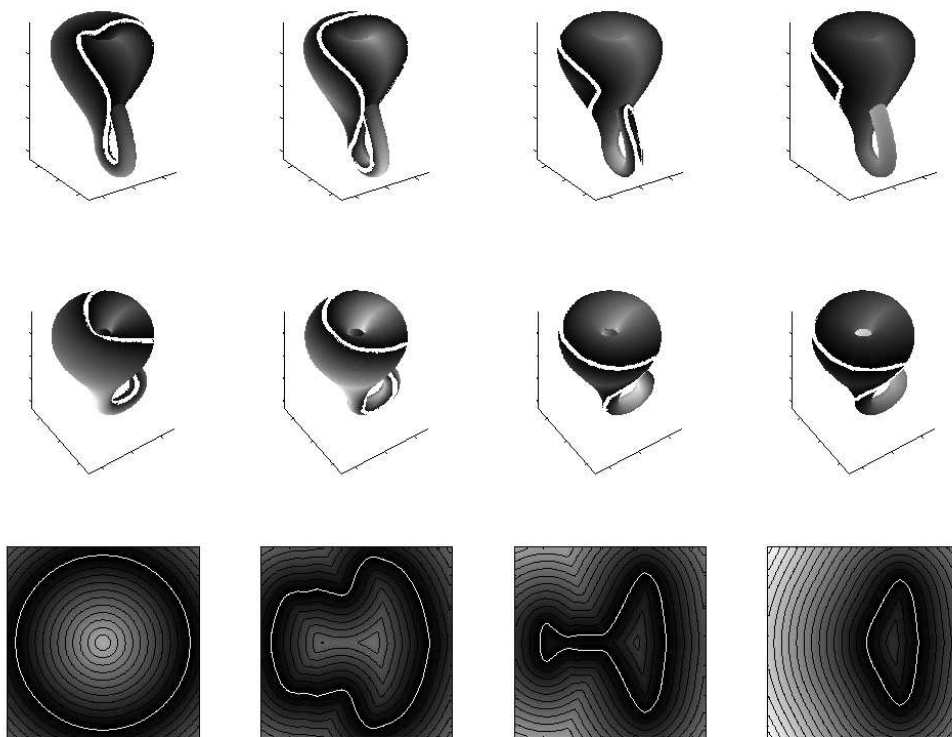


Fig. 3. Geodesic curvature flow on a Klein bottle. Time advances from left to right. Bottom row shows the flow on the parameterization plane.

§6. Simulations and Results

A curve evolution by geodesic curvature flow was implemented on a Klein bottle, see Figure 3. This manifold has high curvature and is self intersecting. The high order numerical scheme, combined with the regularizing process, yields a flow without topological changes of the curve.

The geodesic curvature flow can be applied to images painted on manifolds too. This creates an intrinsic scale space for the images on the manifolds [6]. Here we applied it to the image of the face of a mannequin painted on the mannequin's face manifold, see Figure 4. The face manifold was originally a triangulated manifold, generated by a laser scanner [12]. In order to transform the manifold into a parametric surface, the metric tensor has been approximated from the triangulated manifold. The approximation consisted of matching a second order polynomial patch at every grid point. The matching was done using Singular Value Decomposition (SVD).

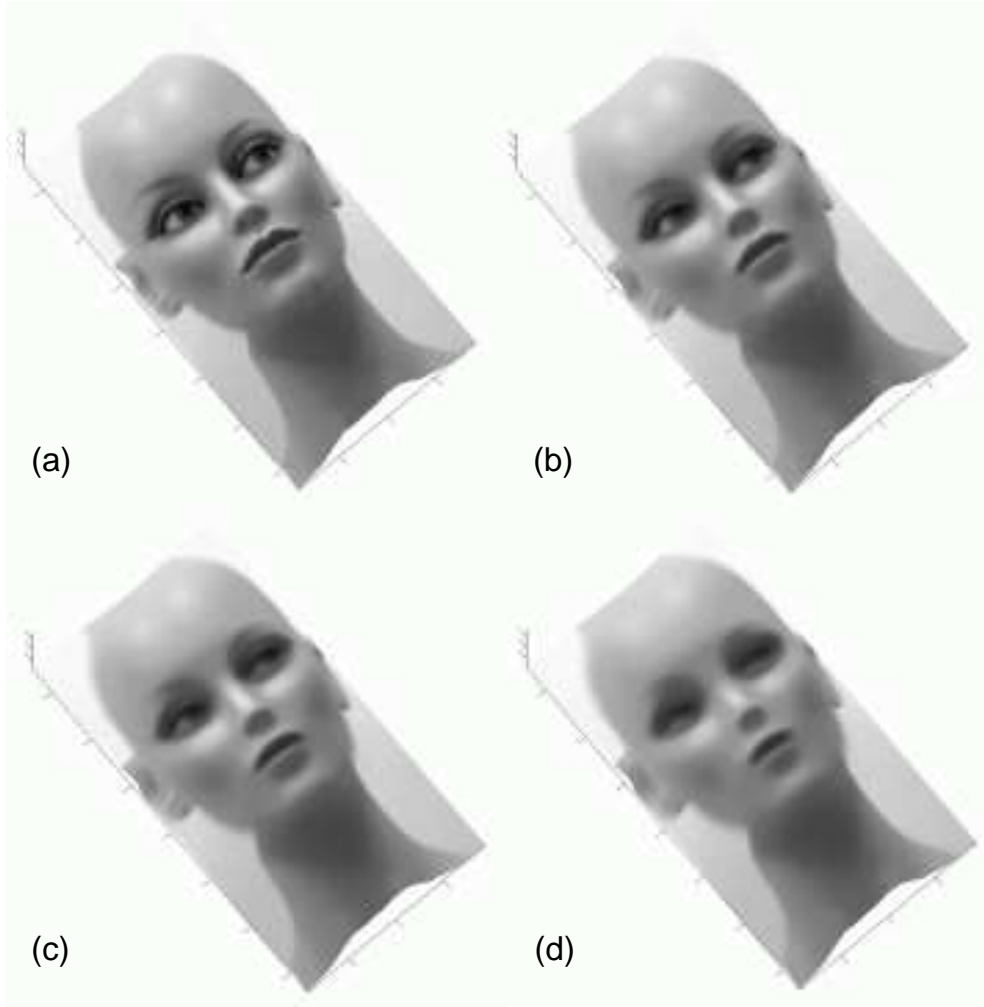


Fig. 4. Geodesic curvature flow of the face image on the face manifold.

§7. Conclusions

We implemented the geodesic curvature flow on manifolds as a flow on the parameterization plane. This approach has several advantages over implementing the flow in the N -dimensional space. An appropriate numerical scheme was devised for implementing this flow with the level sets formulation. The flow of curves and images was demonstrated on a variety of parametric surfaces.

References

1. L. T. Cheng, P. Burchard, B. Merriman, and S. Osher, Motion of curves constrained on surfaces using a level set approach, *CAM Report 00-32*, UCLA (2000).

2. D. L. Chopp, and J. A. Sethian, Flow under curvature: singularity formation, minimal surfaces, and geodesics, *J. Exper. Math.* **2(4)** (1993), 235–255.
3. M. P. Do Carmo, *Differential Geometry of Curves and Surfaces*, Prentice-Hall, New Jersey, USA, 1976.
4. M. Grayson, The heat equation shrinks embedded plane curves to round points, *J. Differ. Geom.* **26** (1987), 285–314.
5. G. S. Jiang and D. Peng, Weighted ENO schemes for Hamilton Jacobi equations, *SIAM J. Sci. Comput.* **21(6)** (2000), 2126–2143.
6. R. Kimmel, Intrinsic scale space for images on surfaces: the geodesic curvature flow, *Graphical Models and Image Processing* **59(5)** (1997), 365–372.
7. R. Kimmel, A. Amir, and A. M. Bruckstein, Finding shortest paths on surfaces using level sets propagation, *IEEE Trans. on PAMI* **17(1)** (1995), 635–640.
8. R. Kimmel, and N. Kiryati, Finding shortest paths on surfaces by fast global approximation and precise local refinement, *Int. Journal of Pattern Recognition and Artificial Intelligence* **10(6)** (1996), 643–656.
9. S. J. Osher, and J. A. Sethian, Fronts propagation with curvature dependent speed: algorithms based on Hamilton-Jacobi formulation, *J. Comput. Phys.* **79** (1988), 12–49.
10. J. A. Sethian, A fast marching level set method for monotonically advancing fronts, *Proc. Nat. Acad. Sci.* **93(4)** (1996), 1591–1595.
11. M. Sussman and E. Fatemi, An efficient, interface preserving level set re-distancing algorithm and its application to interfacial incompressible fluid flow, *SIAM J. Sci. Comput.* **20(4)** (1999), 1165–1191.
12. G. Zigelman and R. Kimmel, Fast 3D laser scanner, Technical report (2001).

Alon Spira and Ron Kimmel
Computer Science department
Technion, Technion city, Haifa, Israel
`salon,ron@cs.technion.ac.il`