Noname manuscript No.
(will be inserted by the editor)

Geodesic Distance and Curves through Isotropic and Anisotropic Heat Equations on Images and Surfaces

Fang Yang · Laurent D.Cohen

Received: date / Accepted: date

Abstract This paper proposes a method to extract geodesic distance and geodesic curves using heat diffusion. The method is based on Vardhan's formula that helps to obtain a numerical approximation of geodesic distance according to metrics based on different heat flows. The heat equation can be utilized by regarding an image or a surface as a medium for heat diffusion and letting the user set at least one source point in the domain. Both isotropic and anisotropic diffusions are considered here to obtain geodesics according to their respective metrics. 1) In the part of the paper where we deal with the isotropic case, we use gray-level intensity to compute the conductivity, i.e. those pixels with gray-levels similar to the source point would have higher conductivity. The model of Perona and Malik, which inhibits heat from diffusing out of homogeneous regions, is also used for geodesic computations in this paper. The two methods are combined and used for more complicated cases. We can also use the norm of the gradient of an image as the feature in the Perona and Malik model to make the heat diffuse along boundaries and edges. 2) For the anisotropic case, we use different eigenvectors and eigenvalues to compose the diffusion tensors to concentrate heat flow along chosen directions. Furthermore, to automate the process of extracting geodesic lines, we propose two automatic methods: a new voting method and a key point method, which are both especially designed for the heat-based method. Our algorithms are tested on synthetic and real images as well as on a mesh. The results are very promising and demonstrate the robustness of the algorithms.

Keywords Heat Diffusion · Geodesic · Minimal Paths · Isotropic · Anisotropic · Automatic Geodesic Extraction · Geodesic Voting · Key Points

F.Yang · L.D.Cohen
Place du Maréchal de Lattre de Tassigny 75775 PARIS Cedex 16
E-mail: yang@ceremade.dauphine.fr, Cohen@ceremade.dauphine.fr

1 Introduction

The heat equation is a partial differential equation that describes the evolution of the distribution of heat (or variation in temperature) in a given region over a certain period of time T. Generally, the form of the heat equation is as follows:

$$\frac{\partial u}{\partial t} - \alpha \Delta u = 0 \tag{1}$$

where α, a positive constant, stands for the thermal diffusivity and Δ represents the Laplace operator. In the physical problem of temperature variation, $u(x,t)$ represents the temperature. More generally, $u(x,t)$ may represent the concentration of a certain substance, like water, whose quantity may vary with time t [22]. Eq.(2) gives a more general form of the heat equation.

$$\partial_t u(x,t) = k(x)\mathrm{div}(D \cdot \nabla u) \tag{2}$$

The coefficient $k(x)$ is the inverse of the specific heat of the substance multiplied by the density of the substance at that location [35]. In the case of a homogeneous isotropic medium, the matrix D has the form of a constant scalar times I_d, where the scalar represents the conductivity and where I_d is the identity matrix. If this scalar value is varying on the domain, we have a general medium. This case is called the isotropic case. In the anisotropic case, the coefficient matrix D has different eigenvalues. This means conductivity is not the same in different orientations.

The heat equation is widely used in many fields. For example, the HKS (Heat Kernel Signature), which is obtained by restricting the heat kernel to the temporal domain, is based on the properties of the heat diffusion process on a shape [14]. In [26], Raviv *et al* used the heat kernel to compute the diffusion distance for shape matching. As introduced in [38] a long time ago, the notion of Scale-Space is based on the Heat Equation. The non-linear heat diffusion can be also used for filtering problem. For example, in [23], Perona and Malik introduced an anisotropic diffusion approach to reduce image noise without removing prominent parts of the image content, and their non-linear diffusion filter only involves scalar diffusion coefficients. In [10,21], Fehrenbach and Mirebeau proposed a non-negative numerical scheme called anisotropic diffusion using lattice basis reduction for image filtering and enhancing. It involves constructing the stencils whose geometry is tailored after the local diffusion tensor.

In this paper, we are interested in the consequences of the work of Varadhan [33], where the author has proposed to approximate the geodesic distance $\phi(p_0, p_x)$ between two points p_0 and p_x on a Riemannian manifold by solving the following equation numerically:

$$\phi(p_0, p_x) = \lim_{t \to 0} \sqrt{-4t \log u_{p_0}(p_x, t)} \tag{3}$$

where u_{p_0} is the heat kernel of Eq.(1), that is, u_{p_0} is a solution with initial value $u_{p_0}(0) = \delta_{p_0}$. This was used recently in [8] to derive a numerical approximation of the geodesic distance, by solving the heat equation numerically with a small time step t. Diffusion is a process of motion of molecules (mass) moving from a place of high density to a place of low density. Based on this, Crane *et al* [8] proposed a method to extract geodesics on surfaces. Intuitively, one may regard the heat diffusion process as a large collection of hot particles moving from the source point p_0 and to the end points p_x, under the assumption that the domain is homogeneous with unit diffusivity $\alpha = 1$. The heat equation is solved based on standard differential operators. Compared with the state-of-the-art methods [31], using heat diffusion to approximate the geodesic distance is computationally more efficient and has comparable accuracy and robustness [8] .

The geodesic distance and geodesic lines on images and surfaces play an important role in computer vision and graphics. They can be applied to vessel segmentation, road extraction, surface remeshing and so on [24]. Generally, the geodesic distance ϕ can be obtained by solving the Eikonal equation numerically using Dijkstra's method [9] or the Fast Marching Method [31,4]. Once we get an approximation of the geodesic distance ϕ, the geodesic lines γ^* between source point p_0 and other points p_x on the domain are extracted by integrating an ordinary differential equation numerically [4,24]:

$$\forall s > 0, \frac{\mathrm{d}\gamma^\star}{\mathrm{d}s} = -D^{-1}\nabla\phi, \gamma^\star(0) = p_x \tag{4}$$

where D is the metric tensor in the anisotropic case. For the isotropic case, $D = \alpha^2 I_d$, and Eq.(4) becomes $\frac{\mathrm{d}\gamma^\star}{\mathrm{d}s} = -\nabla\phi$.

The metrics used in this paper can also be computed by other distance computation techniques such as the Fast Marching Method [31,32,20].

Using a heat method to approximate the geodesic distance has several advantages. The heat method is very fast [8] and also easy to implement. Furthermore, we show that different kinds of features can be extracted by using different diffusion models. Another advantage of this method is that it is not highly sensitive to noise. On the other hand, there are some disadvantages. The heat method is useful within a limited time period. After a long period of time, too much diffusion over the domain will cause blurring and make it hard to sort out the features of interest from all available features.

In this paper, we go beyond the work of Crane *et al* [8] and introduce different heat flows to find the geodesic distance and lines. These flows can be either isotropic or anisotropic, depending on the needs. Note that all the diffusion models used in this paper are linear. In order to extract the geodesic lines automatically, we introduce two new approaches based on geodesic voting and key point detection, which are inspired by [28–30] and [2,12] but adapted to heat diffusion rather than Fast Marching.

This paper is organized as follows: Section 2 presents isotropic and anisotropic heat diffusion models using different potentials and tensors including functions of the image gray values for conductivity, the Perona-Malik (P-M) model, and functions of the image Hessian for the diffusion tensor. We also explain the validity of Varadhan's formula in the cases we consider. Section 3 demonstrates the results of using different diffusion models to acquire geodesic distances and paths. Section 4 introduces two new approaches for automatic geodesic extraction. Section 5 shows experimental results for these automatic methods on images. Section 6 provides some concluding remarks and possible directions for future work.

2 Varadhan's Formula and Heat Diffusion by Different Potentials and Tensors

Heat diffusion comprises 2 types: isotropic and anisotropic. The distinction is made by determining whether the diffusivity is a scalar or a matrix. When we consider heat diffusing on a $N \times N$ image, the initial condition would be:

$$\begin{cases} u_t - \alpha \Delta u = 0, (x, y) \in [0, N] \times [0, N], t \in [0, R^+] \\ u(x, y, t = 0) = \delta_{(x_0, y_0)} \end{cases} \tag{5}$$

where α is a constant in the homogeneous domain and δ_{x_0,y_0} is the dirac distribution centered at $p_0 = (x_0, y_0)$. It should be noted that several source points can be used to diffuse simultaneously.

In Crane *et al*'s method [8], there are three steps to get the geodesic distance ϕ on a surface: 1) compute the heat density: $\partial_t u = \alpha \triangle u$, α is a constant on the whole domain; 2) normalize the gradient: $X = \frac{\nabla u}{|\nabla u|}$; 3) solve the Poisson equation: $\triangle \phi = \mathrm{div}(X)$ to get the distance ϕ. Crane *et al*'s method shows the correlation between the heat density u and the geodesic distance map ϕ. In this paper, we solve the isotropic (or anisotropic) heat equation to get the heat distribution on images where the heat will flow along the direction of a geodesic. Then we apply Eq.(3) directly to get the geodesic distance ϕ.

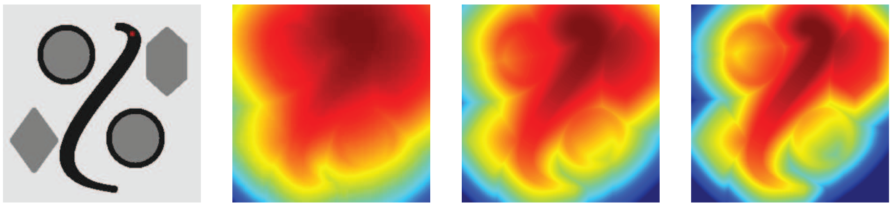
Next, a geodesic curve $\gamma^\star$ between the source point p_0 and another point p_x in the domain can be computed by gradient descent [4,3], by using $\frac{\mathrm{d}\gamma^\star}{\mathrm{d}s} = -\nabla \phi$. This backtracking becomes Eq.(4) in the general anisotropic case.

2.1 Isotropic Diffusion

Four situations of isotropic heat diffusion are discussed here: 1) conductivity based on a function of the gray level, 2) P-M model [23] (although Perona and Malik claimed that their model is anisotropic, we still consider it isotropic following [15], since they use a scalar diffusivity and not a tensor), 3) the combination of conductivity and the P-M model, 4) a P-M model using the norm of gradient of the image as the feature.

2.1.1 Conductivity

In Eq.(1), α represents thermal diffusivity. In the homogeneous case, α is a constant. Putting the source point in the center of the image, heat will diffuse across concentric circles and geodesics will be straight lines orthogonal to these



(a) (b) (c) (d)

Fig. 1 The example of how heat propagates on a synthetic image and the effect of the power n in (6), (a) is the original 300×300 image composed of several parts including a wide black curve in the middle and its surroundings, the red point which is the source-point $(187, 36)$manually set on top of the curve from which heat can diffuse, (b) is the heat distribution for $n = 1$ in (6), (c) is the result for $n = 2$ in (6), and (d) is the result for $n = 3$ in (6). These results are generated in the same period of time T.

circles. We are motivated by finding geodesic paths that follow related gray-level values, and therefore use a conductivity which is a function of the gray-level at each pixel. Given a source point p_0 on an image f, the conductivity of p_0 equals to 1. Point p_x has a higher conductivity α when the gray-value difference between p_0 and p_x is small as shown in Eq.(6).

$$\alpha_{p_x} = |(1 - |f(p_0) - f(p_x)|)|^n + \varepsilon \tag{6}$$

$n = 1, 2, 3...$, ε is a small positive constant that prevents α from vanishing. The value of n depends on the contrast between the interesting features in the image and the image background. In other words, if there exist fewer differences between the interesting features and their background, we can set a higher n. From Fig.1, it can be clearly seen that the black wide curve (the part to be enhanced) is the most visible in (d), compared with the other two results (b) and (c).

2.1.2 Perona-Malik (P-M) model

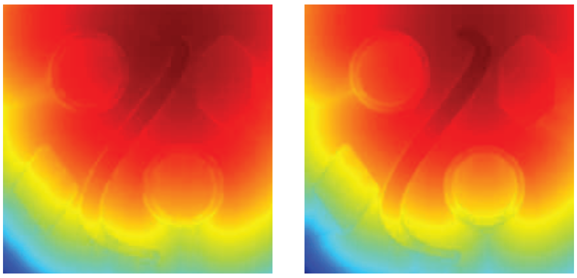
As stated above, the P-M model [23] is not a real anisotropic model because D used in (7) is a scalar and not a tensor.

$$\begin{cases} \frac{\partial u}{\partial t} = \operatorname{div}(D\nabla u) \\ u(x, y, t = 0) = \delta_{(x_0, y_0)} \end{cases} \tag{7}$$

There are two forms of D usually used, both being positive decreasing functions of the gradient, which are given by

$$D = e^{-(\|\nabla f\|/K)^2} \quad \text{or} \quad D = \frac{1}{1 + (\|\nabla f\|/K)^2} \tag{8}$$

where K is the contrast parameter and $\|\nabla f\|$ is the norm of the gradient of the image. Diffusion processes in Sect.2.1.1 tend to equilibrate the concentration differences in the materials, while the feature (Eq.(8)) used in the P-M model can constrain the diffusion process inside the homogeneous regions. From



(a) (b)

Fig. 2 Example for showing effect of the item $\nabla D \cdot \nabla u$, (a) is the result without the item, and (b) is the result with the item.

Eq.(8) we can see that wherever there is higher gradient there is lower diffusivity, which indicates that P-M model inhibits heat from leaking outside a homogeneous region. Note that since the goal here is different from the usual P-M equation where the heat density is the image f itself, the initial value of u here is different from the usual case. We use a Dirac distribution as the initial density. Eq.(7) can be also written as Eq.(9)

$$\frac{\partial u}{\partial t} = D\varDelta u + \nabla D \cdot \nabla u \tag{9}$$

Compared with Eq.(1), there exists an additional first derivative term in Eq.(9): $\nabla D \cdot \nabla u$. Fig.2 shows the experimental result on a synthetic image. From this result, it can be seen that there is a difference between (a), the form the result takes without adding the item, and (b), the result with the item considered. Compared to (b), we can see that (a) has more heat around the edges, which shows that adding this item $\nabla D \cdot \nabla u$ helps to restrain the heat from leaking out of a region slightly.

2.1.3 Combination of Conductivity and P-M Model

As we can see above, both methods have their own advantages. Using conductivity is direct and useful in simple scenes, but it becomes insuffcient when dealing with more complicated scenes. The P-M method helps to weaken heat diffusion on the edges and boundaries. Thus it can be used as an auxiliary factor. This is the reason why we combine the two methods together:

$$\frac{\partial u}{\partial t} = \alpha \cdot \mathrm{div}(D\nabla u) \tag{10}$$

Here, α is defined by Eq.(6). Thus diffusion depends on both region and edge based features. The advantage can be seen in Fig.3 where the heat becomes more concentrated along the central curve as a result of this combination.

Another advantage of using this combination is that it can get the centerline automatically. According to Eq.(3), it is indicated that wherever there is a

(a) (b) (c)

Fig. 3 Example on the same synthetic image showing the combination of conductivity and P-M diffusivity. A source point settles on top of the curve. After the same period of time, (a) is the result of using the cubic form of Eq.(6), (b) corresponds to the PM model Eq.(8), (c) is the result of using the combination of conductivity and P-M model Eq.(10).

larger heat density there is a smaller distance between the points on the image and the source point p_0. By using Eq.(10), heat will be mostly concentrated in the center of the region containing the source point. Fig.8 gives an example of centerline extraction in a vessel image.

2.1.4 A P-M Model that Follows the Edges

In fact, besides the features in Eq.(8), there are other features that can also be used in the P-M model. Contrary to the features in Eq.(8) that keep the heat inside a region, we propose to define features that help heat focus on the edges or boundaries by enhancing the potential on the edges:

$$\begin{cases} \partial_t u = \operatorname{div}(D\nabla u) \\ D = \|\nabla f\|^2 + \varepsilon \end{cases} \tag{11}$$

In this model, heat diffuses faster wherever the gradient gets higher such as on an edge or across thin structures. In order to keep the diffusion coefficient strictly positive, we add a small positive constant ε to $\|\nabla f\|$. Examples are shown in Fig.4. In the row above, without enhancing $\|\nabla f\|$, the very thin line can hardly be taken into account, as shown in (b), while in (c), the heat travels mostly along the thin curve in the middle, using Eq.(11). In addition, a much wider line of interest and its background with several polygons is shown in (d). Both (e) and (f) are the results of diffusion using Eq.(11) in P-M model where the diffusion starts from two different positions of the source point. When the heat starts diffusing from the red point, it almost goes along the boundary of the hexagon and then heat diffuses to the curve. The heat also concentrates on the boundary of the curve. The same phenomenon can be seen in (f) where heat starts diffusing in the center of the curve and then goes along the double edge. Thus Eq.(11) is good at extracting features such as edges and boundaries.

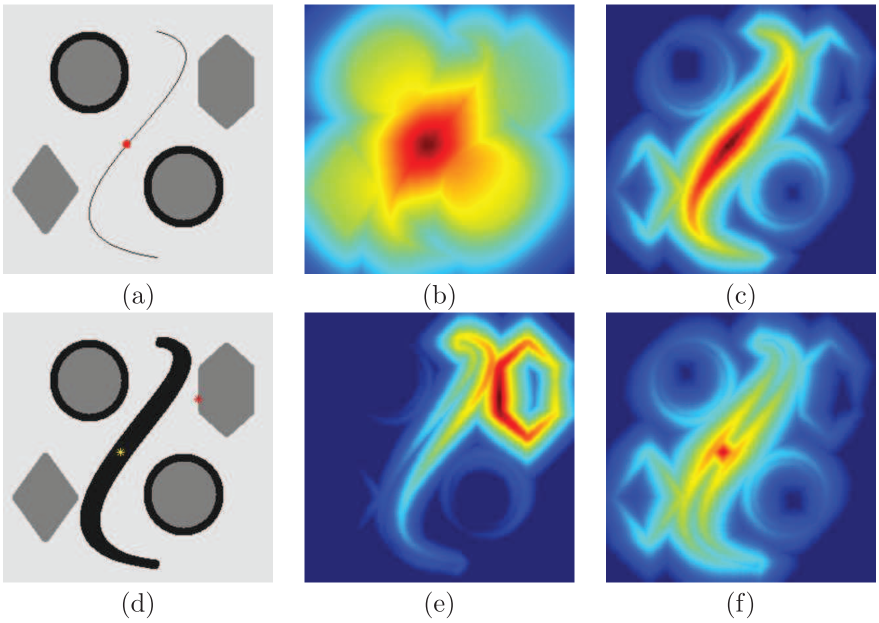


Fig. 4 Example on a synthetic image to illustrate how Eq.(11) works. The source point is placed on the very thin line in (a) and a wider curve in (d), the red point in (a) is the source point for (b) and (c). (b) is the result of diffusing on (a) by a method discussed in (6), (c) is the diffusion result of (a) using (11). The red point on (d) indicates the position of the source point for (e) and the yellow point indicates the position of the source point for (f). Both (e) and (f) are the diffusion results of (d) generated by using $|\nabla f|^2$ as the feature in the P-M model.

2.2 Anisotropic Diffusion

Anisotropic diffusion has a form as follows:

$$u_t = \mathrm{div}(D\nabla u) \tag{12}$$

where D is a diffusion tensor rather than a scalar. It is a tensor field of symmetric positive matrices that encodes the local orientation and anisotropy of an image. This anisotropic diffusion makes heat propagate in the direction that we design [3,13] by defining the relevant tensor D. Weickert [37] proposed a coherence enhancing diffusion method, using a nonlinear anisotropic diffusion equation for filtering problems. A symmetric and positive definite diffusion tensor is used in this method. It is obtained by the tensor product of ∇f: $J_\rho(\nabla f_\sigma) := K_\rho * (\nabla f_\sigma \otimes \nabla f_\sigma)$, where K_ρ is a Gaussian kernel. In [37], the eigenvectors are the same as in $J_\rho(\nabla f_\sigma)$ and the eigenvalues are chosen to make diffusion act mainly along the direction with the highest coherence. An improved structure tensor is proposed in [18] to get an integrated edge and junction detection method. The structure tensor is calculated by means of Gaussian derivative filters of the image f. The authors proposed multiple ways to improve the structure tensor including using a higher sampling rate, improving corner localization etc. Generally, the gradient ∇f of the image is usually taken into account to measure the local direction of edges or texture [24]. In [3], the authors proposed an interactive vessel segmentation method to extract the centerlines as well as the boundaries of the vessels. In this method, they defined the metric using the eigenvectors and eigenvalues obtained from OOF (optimally oriented flux) [19]. This metric is oriented along the estimated direction of the vessel, allowing a higher velocity on the centerline and with an estimate of the vessel local radius. Since we do not segment the boundaries in our paper, we just use the Hessian matrix to construct the metric.

2.2.1 Eigenvalues and Eigenvectors

The tensor field can be diagonalized as in [24]:

$$T_x = \lambda_1(x)e_1(x)e_1(x)^T + \lambda_2(x)e_2(x)e_2(x)^T \tag{13}$$

The normalized vector fields $e_i(x)$ are orthogonal eigenvectors of the symmetric matrix T_x, and the $\lambda_i(x)$ are the corresponding eigenvalues, with $0 < \lambda_1(x) \leq \lambda_2(x)$. Following [24], the anisotropy $A(x)$ is defined as:

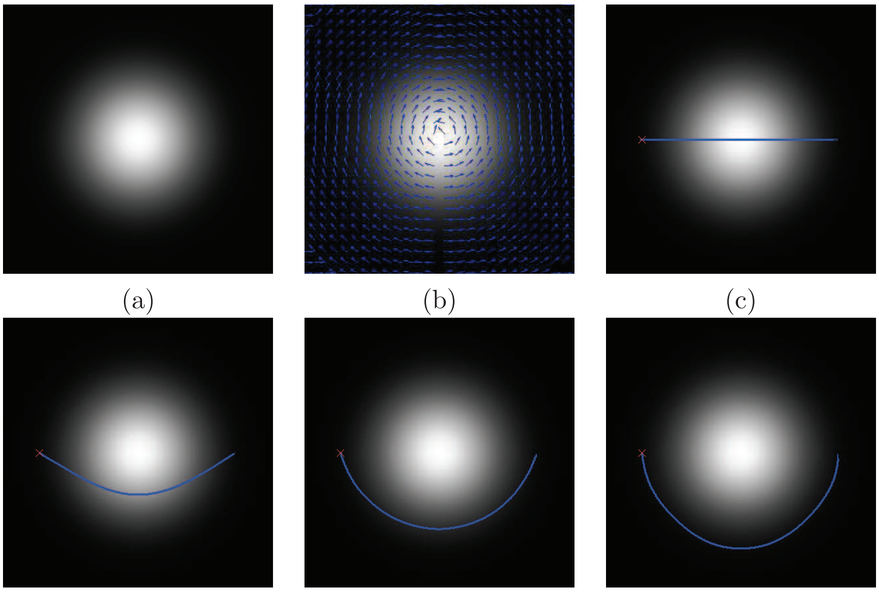
$$A(x) = \frac{\lambda_2(x) - \lambda_1(x)}{\lambda_2(x) + \lambda_1(x)} \tag{14}$$

When $\lambda_1(x) = \lambda_2(x)$, the anisotropy $A(x)$ is 0, and the tensor is in fact a scalar metric which makes geodesics the shortest paths according to the isotropically weighted distance.

2.2.2 Anisotropic Diffusion Tensor

When $\lambda_1 \neq \lambda_2$, it is anisotropic. As mentioned before, $\lambda_2 \geq \lambda_1$, λ_2 controls the direction of the heat flow. When λ_2 is far larger than λ_1, the heat flows in the direction of e_2 while only a little goes into the orthogonal direction. Fig.5 depicts the effect of the change of anisotropy in detecting a shortest path. We define the gradient direction of the image as e_1, which is the radial direction, and its orthogonal direction as e_2 (in fact, the eigenvectors are the same as those used in [37]). When $\lambda_1 = \lambda_2$, the heat diffusion begets the Euclidean distance map, and the shortest path is a straight line. As the anisotropy grows, the direction of heat flow travels more along the tangent direction and the geodesic lines get closer and closer to a half circle.

In [11], the authors introduced a multi-scale vessel enhancement method by interpreting geometrically the eigenvalues of the Hessian matrix. Using the Hessian eigenvectors, the local orientation of the image can be estimated, allowing to find out where there are tubular structures like vessels. Here we use a fixed Gaussian kernel and compute the 2D Hessian matrix, then compute the eigenvalues and eigenvectors. We use the eigenvectors as a tensor field and control the anisotropy.



(a) (b) (c)

(d) (e) (f)

Fig. 5 (a) a Gaussian image, (b) tensor field by gradient, (c) to (f) are the shortest paths between the two user-chosen points, the corresponding anisotropies are 0, 0.5, 0.8 and 0.99

Fig.6 shows the isotropic and anisotropic heat diffusions respectively on a U-tube image. The U-tube image is given by (a). (b) is the distance map obtained by using conductivity Eq.(6), $n = 3$. (c) is the distance map obtained by using anisotropic diffusion. (d) shows the tensor field that is used in the anisotropic diffusion, and it is obtained by Hessian matrix. (e) is the geodesic line obtained by backtracking in (b), we can see that the geodesic line takes a shortcut. When the heat diffuses by taking the local orientation into account, this shortcut is avoided. (f) shows geodesic line by backtracking in (c). As seen, the heat travels along the tensor field, and by backtracking, the line is exactly located on the tube in the correct direction.

2.3 Heat Diffusion on Meshes

The heat equation on meshes is similar to the one on images though the heat is transferred from one vertex to another, not from pixel to pixel. We introduce a numerical method to solve the heat equation on triangle meshes. The mesh is composed of faces $\{f_m\}_{1\leq m\leq M}$ and vertices $\{v_n\}_{1\leq n\leq N}$ where M and N are the numbers of faces and vertices respectively.

First, we need to compute the cotangent Laplacian W of the mesh. It can be obtained by using Eq.(15) [25,27]:

$$W_{i,j} = \cot(\alpha_{i,j}) + \cot(\beta_{i,j}) \tag{15}$$

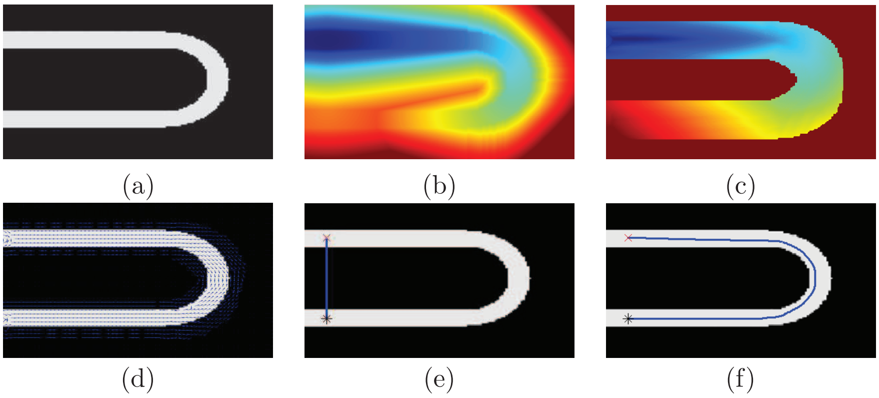


Fig. 6 Experiment on a U-tube structure: (a) original image, (d) tensor field, (b) and (c) are the distance maps obtained by isotropic and anisotropic diffusions respectively, (e) and (f) are the corresponding geodesic lines.

where $\alpha_{i,j}$ and $\beta_{i,j}$ are the two angles opposite to the edge (v_i, v_j), which connect two vertices. Next, we need to compute the symmetric Laplacian matrix $L = D - W$, where $D = \text{diag}_i(\sum_j W_{i,j})$. At last, we get the normalized operators $\tilde{W} = D^{-1}W$ and the Laplace operator is $\tilde{L} = D^{-1}L$.

The heat diffusion on a mesh solves: $u_t = -\tilde{L}u$. Here the conductivity of the domain is assumed constant. When it comes to enhancing other special features, for example, the curvature in order to extract the edges on mesh, or the texture on the surface in order to find characteristic lines on the surface, we incorporate these features into the heat equation. It yields:

$$\frac{\partial u}{\partial t} = -\tilde{L}(u * P) \tag{16}$$

where P plays a role similar to heat conductivity α in (6) and determines the evolution and distribution of heat on the surface.

2.4 The Applicability of Varadhan's Formula

As explained in the introduction, the main relation between heat diffusion and distance maps comes from the Varadhan's formula. In this section, we give a closer look to this formula and its extensions.

In [6], the authors introduced approximate solutions for the Green function of uniformly parabolic second-order operators with variables, by using expansions. Let L be the general Laplacian operator which comprises the second and first order terms:

$$Lu := \sum_{i,j}^{n} a_{i,j}(x)\frac{\partial^2 u}{\partial x_i \partial x_j} + \sum_{j}^{n} b_j(x)\frac{\partial u}{\partial x_j} + c(x)u \tag{17}$$

Our heat equation with the boundary condition becomes:

$$\begin{cases} \partial_t u - Lu = 0, & \text{in}(0,\infty) \times \mathbb{R}^N \\ u(x,0) = \delta_{x_0}, & \text{on}\mathbb{R}^N \end{cases} \tag{18}$$

According to [34,36,6], the operator L can be interpreted as a Laplace-Beltrami operator on a manifold with lower order terms, thus, we can obtain the Green function (or fundamental solution, heat kernel) $\mathcal{G}_t(x,y)$ of Eq.(18) represented by an asymptotic expansion of the form:

$$\mathcal{G}(x,y,t) = \frac{e^{-\frac{\phi(x,y)^2}{4t}}}{(4\pi t)^{N/2}} (\sum_{k=0}^{\infty} \mathcal{G}^{(k)}(x,y)) \tag{19}$$

as $t \to 0^+$, where $\phi(x,y)$ is the geodesic distance induced by a Riemannian metric derived from the coefficients $\{a_{i,j}\}$ between points x and y, $\mathcal{G}^{(k)}(x,y)$ are smooth functions.

The fundamental solution of Eq.(19) satisfies the formula introduced by Varadhan [33]:

$$\lim_{t\to 0}[-4t \log u_x(y,t)] = \phi^2(x,y) \tag{20}$$

where here $u_x(y,t) = \mathcal{G}(x,y,t)$. For example, this formula is well understood in the case of the homogeneous heat equation posed on the whole domain $\mathbb{R}^2$. Then, the Green function is in fact a Gaussian function and the explicit solution is

$$u_x(y,t) = (2\pi t)^{-k/2} \exp\{-\frac{1}{4t}\|x-y\|^2\} \tag{21}$$

It can be easily seen that this function satisfies formula Eq.(20).

In this paper, we introduce three kinds of heat flows.

1. The heat equation for the conductivity case is obtained from Eq.(18), taking $a_{i,j}(x) = \alpha_{p_x}\delta_{i,j}$, where α_{p_x} is defined in Eq.(6) and $\delta_{i,j}$ is the kronecker symbol, equal to 1 or 0 depending on i equal to or different from j.
   And we have $b_j(x) \equiv 0$, $c(x) \equiv 0$. Therefore, the distance ϕ is a weighted distance. The geodesic paths correspond to minimal paths relative to isotropic potential equal to the conductivity α_{p_x}.
2. For the P-M case, of section 2.1.2, we have $a_{i,j}(x) = D(x)\delta_{i,j}$ and $(b_j(x))_j = \nabla D$, $c \equiv 0$ in Eq.(18).
3. For the anisotropic case, of section 2.2, we can take, by developping $\text{div}(D\nabla u)$ from Eq.12, $a_{i,j}(x) = D_{i,j}(x)$, $b_j(x) = div((D_{i,j}(x))_i)$ and $c(x) \equiv 0$. Since for the general anisotropic case the distance ϕ derives from a Riemanian metric, the orientation of the geodesic lines have to agree as much as possible with the eigenvectors of the metric.

In all cases above, Varadhan's formula means that the corresponding heat flow allows to find an estimate for the distance map according to the Riemannian metric derived from the coefficients $\{a_{i,j}\}$.

3 Experiments and Analysis of Different Heat Flows

3.1 Experiments Data and Settings

We test the different heat diffusion models on several datasets:

Isotropic diffusion on real images. In the experiment of road extraction (Fig.7), the data is an optical remote sensing image and is resized to 300×300. In the experiment of vessel extraction (Fig.8), the image size is 512×512. The conductivity is given by Eq.(6), and $n = 3$, the coefficient K in Eq.(8) is 0.03.

Isotropic diffusion on a noisy image. Fig.9 is an example on a noisy image (a), with a given percentage of corrupted pixels, ξ equals to 0.133.

Anisotropic diffusion on a synthetic images. In the spiral experiment in Fig.10, we use the Hessian matrix to define the diffusion tensor and the anisotropy is set to 0.9.

Isotropic diffusion on a mesh. The block of fig.11 has 57184 faces and 25894 vertices. Curvature in the mesh was computed following the method in [5] and [1].

In addition, it should be guaranteed that the heat has spread all over the domain (i.e. the image), where the structures to be extracted are all included. On the other hand, the heat is not supposed to propagate for a long time because the image will eventually get blurred. In order to achieve that, we must manage to set an iteration number in accordance with the problem data (size of the image, position of the source point) while managing control of the heat flows. Therefore, the diffusion has to take place during a limited and short period of time. In addition, to make sure that the values are computed within a reasonable number of time iterations, the time step of each iteration τ in $\frac{\partial u}{\partial t} - \tau \cdot \alpha \Delta u = 0$ is set to $\tau = 0.2$, which satisfies the CFL condition [7].

3.2 Results and Analysis

To evaluate the performance of the centerline extracted by different methods, we compute the *precision* and *recall* criteria given by the following formula:

$$\begin{cases} recall = \frac{TP}{TP+FN} \\ precision = \frac{TP}{TP+FP} \end{cases} \tag{22}$$

where TP is the length of extracted centerline that matches the manually labeled ground truth, FP represents the length of extracted centerline which are not on the ground truth, and FN is the length of the ground truth but that is not extracted.

Isotropic diffusion on real images.

In Fig.7, the source point is set on the top of the very thin white line (which is a road)with the endpoint at the bottom. The blue curves in (a) to (d) are the paths extracted by backtracking from the end point to the source point.

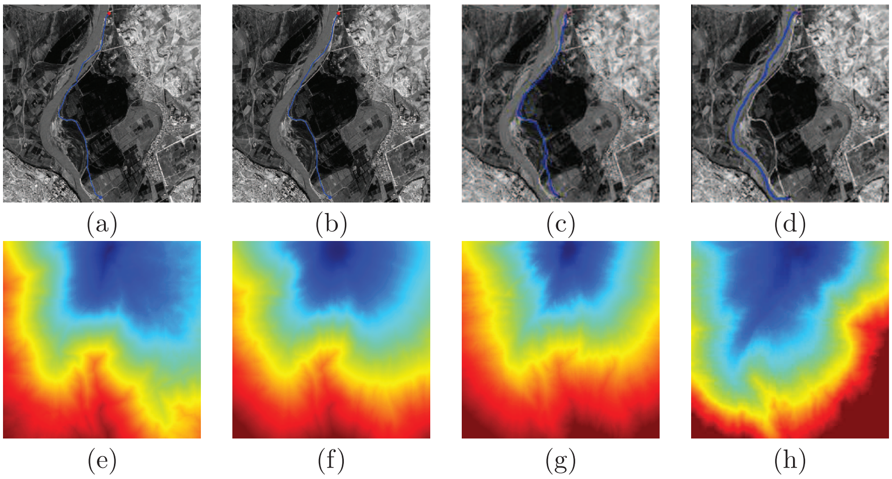


Fig. 7 Experiment on the image of a road: the top row from left to right displays the corresponding path extracted by using Fast Marching, the heat method by using respectively Eq.(6) as conductivity, $n = 2$ and $\epsilon = 0$ Eq.(11) and Eq.(8) in the P-M model, the bottom row displays the distance maps by using these methods respectively.

It is based on the distance maps obtained respectively by using isotropic Fast Marching, by using the heat method with conductivity Eq.(1), by using P-M diffusion model with Eq.(11) and by using P-M model using Eq.(8) as shown from (e) to (h). In (a) and (b), we use the same metric for the Fast Marching Method and heat diffusion.

From Table 1, the road extracted in (a) and (b) has a similar recall and precision. But the road extracted in (b) is much smoother than the one from Fast Marching (a). In addition, the P-M model based on Eq.(11) is good at extracting thin structures, see (c). Futher, in (d), by using Eq.(8) in the P-M diffusion model, the more homogeneous parts can be easily distinguished. This indicates that the heat diffusion by the use of Eq.(8) in the P-M model is likely to present good results when there is a relatively larger part to be extracted. Moreover, heat in such a case is easily diffused in places where a few changes exist.

The experiment of vessels is shown in Fig.8, in which the source point (marked as red cross) and end points (marked as black) are given by the user. By using isotropic Fast Marching (a), the extracted lines do not exactly follow the centerline, especially in the center. By using the isotropic linear heat diffusion Eq.(1) with Eq.(6) as the conductivity, due to directionally independent heat scattering, the centerline of the vessels is not completely right. (c) is the result that combines the P-M model with the conductivity Eq.(10), using Eq.(8) in P-M model, the heat is more concentrated in a homogeneous area where the source point stays. From (d), we can see that the centerlines are well extracted despite the position of the manually set source point, which means that the point of most concentrated heat moves to the centerline in the process of diffusion. In this test, compared to the center line ground truth, it can be seen in Table.1 that Eq.(10) is effective in extracting the center line of the vascular-like structure.

Table 1 (the indexes of evaluation%).

data	method	recall	precision
Road	Fast Marching	79.59	72.87
Fig.7	Heat Diffusion(6)	76.32	71.00
	Heat Diffusion(11)	98.11	91.00
Vessels	Fast Marching	84.73	63.68
Fig.8	Heat Diffusion(6)	91.78	67.01
	Heat Diffusion(10)	92.26	70.09

Isotropic diffusion on a noisy image

The diffusive nature of the heat equation causes instant smoothing. Even if there is a temperature discontinuity at initial time $t = t_0$, the temperature becomes smooth as soon as $t > t_0$. Solutions of the heat equation are characterized by a gradual smoothing process from the initial temperature distribution by the flow of heat from warmer to colder areas of an object, and this can be considered as a blurring process. This is why the heat equation is used for filtering problems. And also in our case, the geodesic curves that are extracted are not so affected by noise. Fig.9 is an example on a noisy image (a), with a percentage 0.133 of corrupted pixels. Given a source point and an end point on both ends of the black curve, the red lines on (b) and (c) are obtained by the isotropic Fast Marching Method and heat method using P-M model respectively. The potential used in Fast Marching and the scalar used in P-M model are the same. Both are the norm of gradient of the image. From the result we can see that the heat method gives a better result than Fast Marching despite the noise, which indicates that the heat method is less sensitive to noise.

Anisotropic diffusion on a synthetic images

Fig.10 illustrates an experiment using anisotropic heat diffusion. The results are as follows: (a) is the original spiral image, (d) is its corresponding tensor field by using the Hessian matrix. (b) is the distance map obtained via the isotropic heat diffusion (Eq.(1) as the diffusion model, and Eq.(6) as the conductivity with $n = 3$) and (c) is the distance map obtained via the anisotropic heat diffusion, (e) and (f) are the extracted lines. From the results, we can see that in (b) and (e), using isotropic diffusion, the temperature blurs in the process of diffusion, and the path extracted takes the shortcut from the end point to the source point, while in (c) and (f), using anisotropic diffusion, the path backtracks along the spiral line, and the heat diffuses predominantly along the spiral apparently.

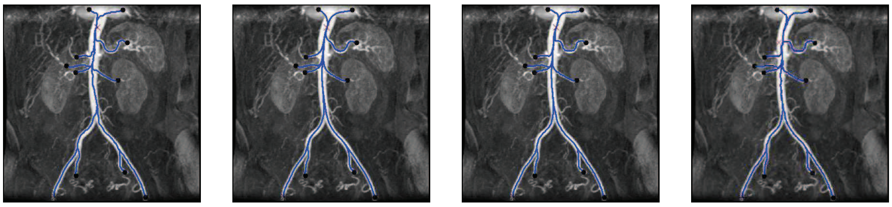


Fig. 8 Experiment on real vessel image: the red cross is the manually set source point, the black spots are the end points provided by the user, and the blue curves are the extracted geodesics (a) is the result by isotropic Fast Marching (b) is the result by only using the conductivity (Eq.6), (c) and (d) are the results by using the combination (Eq.10), but with different source points.

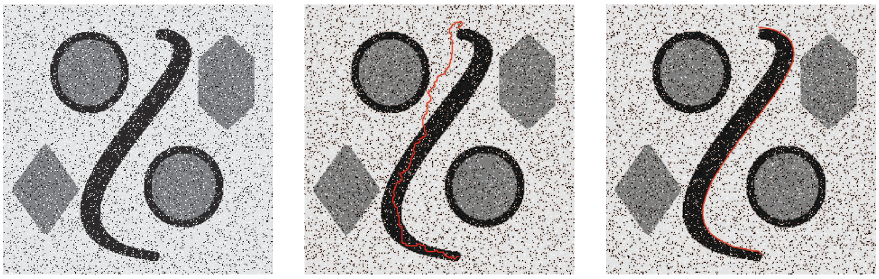


Fig. 9 Experiment on a noisy image (a), the red line on (b) is obtained by Fast Marching Method, the red line on (c) is the result of P-M heat method (11).

Isotropic diffusion on a mesh

To illustrate the method of section 2.3, Fig.11 uses the curvature of the block as diffusivity. Furthermore, the higher the curvature at a point, the higher the probability that at this point the amount of heat received is larger than at the neighboring points: (a) is the original block structure; (b) is the distance map, as shown in (b), the distances on the edges are smaller than the flat surfaces, which means heat is more concentrated on the edges of the block, in comparison with the smooth and flat parts. (c) shows the result of the minimal paths between the ten end points and three source points. It is very conspicuous that that all paths go along the edges.

As is introduced above, the heat flows are not strongly affected by noise. For the different heat flows, there are other advantages and disadvantages which we list in Table.2, where IHF and AHF are abbreviations for the isotropic heat flow and anisotropic heat flow.

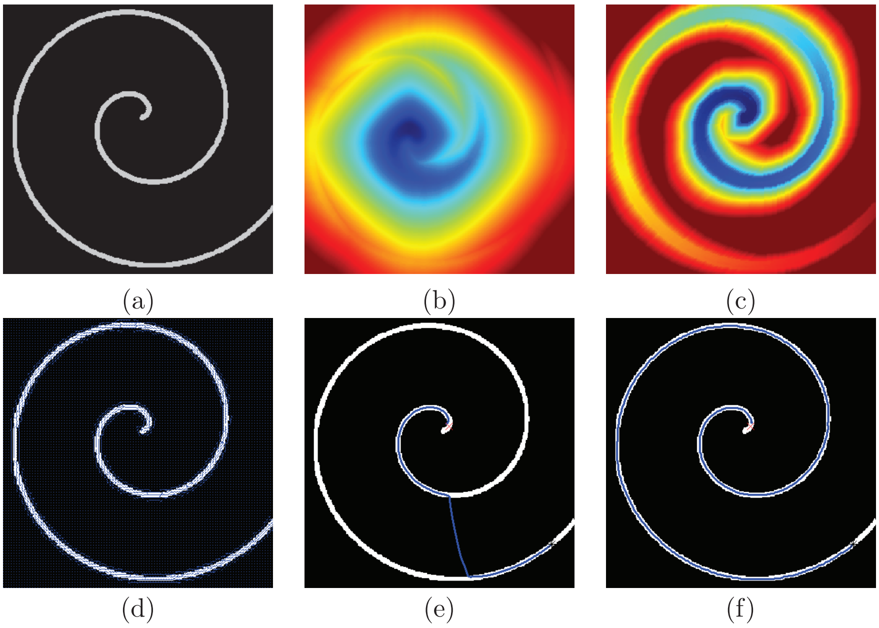


Fig. 10 Experiment on a spiral: (a) original spiral image; (d) tensor field; (b) distance map by isotropic heat diffusion; (e) geodesic line extracted corresponding to (b); (c) distance map by anisotropic heat diffusion; (f) geodesic line extracted corresponding to (c).

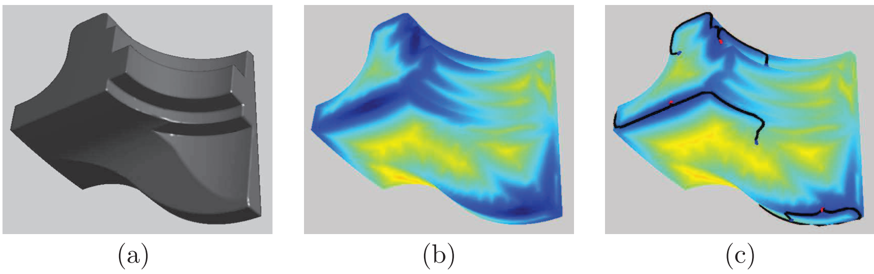


Fig. 11 Experiment on the wedge-like block, three points on the edges are chosen to be the source points, and 10 points are randomly chosen as the end point. (a) is the data, (b) the distance map, (c) the paths extracted along the edges of the block.

4 Automatic Segmentation Based on Heat Diffusion

In the previous sections, we introduced how to obtain the geodesic distance and curves by using different heat diffusion equations. However, in all of these methods, the user has to provide at least one source point and several end points to extract the curves. This extensive manual intervention makes it very tedious to extract complex curves. To address this issue, we propose two algorithms for extracting the geodesic lines automatically without having to provide the end points.

Table 2 (Comparison of different flows).

	Metrics	Advantages	Disadvantages
	Conductivity	$\surd$ Convenient Intuitive	$\times$ Complicated scenes
IHF	P-M Eq.(8)	$\surd$ Homogeneous regions	$\times$ For edges
	P-M Eq.(11)	$\surd$ Edges Boundaries	$\times$ For regions
AHF		$\surd$ Specified directions	$\times$ Time Consumption

The first algorithm is realized by the extension of the method of geodesic voting, which was proposed in [29]. The authors first use the Fast Marching Method to get the distance map, and subsequently use the boundary of the image domain, or randomly selected points, as the end points. Backtracking from these end points to the source point, there will be numerous paths. At each point of the image domain, the geodesic density is defined as the number of paths that go through that point. By thresholding the density, an automatic segmentation of the desired structures can be obtained. The general idea of our voting method is that we set some time $\Delta T > 0$ for the diffusion, which depends on the size of the image. After time ΔT, heat can pervade a certain region surrounding the source point p_0. The pixels on the front (boundary) of this region are then used as the end points for backtracking to p_0. Then, setting a cutoff-threshold ϵ on the geodesic density, we retain only those pixels, which have geodesic densities above this threshold.

The second method is inspired by the key point method described in [2]. In this method, a set of contour curves or thin structures are obtained as a set of minimal geodesic paths connecting successive keypoints. These keypoints are defined in an iterative way by selecting the first point on the Fast Marching front for which the minimal path reaches a given curve length. We refer to [2] for the rationale and details on the method. Here, in our own method, we first let the heat diffuse for some time ΔT_1 and use the current source point as the center and r as radius to form a circle. We identify the points on the circle with the largest temperature values (note that there might be more than one large value, and we are looking for the peaks of the heat density larger than a prescribed threshold). The new source points are located at these peak points. After obtaining the new source points from the peak points, we let them begin to diffuse one after the other until a stopping criterion is met. This is a particular topic of discussion in the keypoint section of the paper.

4.1 Voting Method

In [29], the authors present a novel method for automatic segmentation of tree structures, named geodesic voting. First, the authors obtained the distance map by using the Fast Marching Method, then they use some endpoints chosen automatically to backtrack to the source point. Thus there will be a series of paths extracted. The points located on these paths can be used to define the geodesic density:

$$\mu(p) = \sum_{n=1}^{N} \delta_p(l_n) \tag{23}$$

where $\delta_p(l) = 1$ if pixel p is crossed by path l, N being the number of paths. A threshold-cutoff for the geodesic density is also set. We retain only the pixels with a number of paths above the prescribed threshold in the final result.

In this section, we introduce three voting methods. They will be detailed in the ensuing subsections. One method is to vote from the front of heat distribution directly. As stated above, the paths are extracted from the boundary of the heat distribution after time ΔT. They consist in joining each end point to the source point by backtracking. Fig.12 shows how the time ΔT affects the results. In the first row, we choose a smaller time $\Delta T'$, which is half of the ΔT in the second row. (a) and (d) show the heat distribution after $\Delta T'$ and ΔT and the blue curve surrounding the regions are the fronts which are considered as the end points. (b) and (e) are the paths obtained by backtracking from the front points to the source point. (c) and (f) give the structure extracted by voting. Clearly we can see that in Row2, the method gives a more thorough result compared to Row1, which emphasizes the importance of how we choose ΔT.

Another voting method is to add new voting points within a smaller period time ΔT_1. We then track back from the front to the source point and vote for the first result. Next, we let the heat continue to diffuse for another time ΔT_1 to track back and vote again, adding the new result to the first result. This process of letting the heat diffuse within a period of time ΔT_1 is reiterated as many times as need be until the stopping criterion is fulfilled.

The third voting method is realized by resetting the source point. After time ΔT_1, the first result can still be obtained by voting. Then, we use these points as the new source points, and let the heat diffuse for another ΔT_1. The new results then add-up as the collection of source points for the following diffusion. This process is repeated until the stopping criterion is met. Experiments of these three voting methods are compared in the following section **??** dedicated to the experiments.

4.1.1 Voting from the Front

First of all, we set a source point p_0 at the location of interest. After $\Delta T > 0$, the distance map D is obtained by Eq.(3). The positions on the boundary of the region covered by heat are considered as end points. By backtracking from these end points back to the source point, we get many geodesic lines. Setting up an appropriate threshold ϵ for the geodesic density β can be useful for getting the right path. Generally, we choose $\epsilon = \max(\beta)/M$, where M is a constant. For example, in Fig.12(f), M equals 30 in this case, but M can be different in different cases.

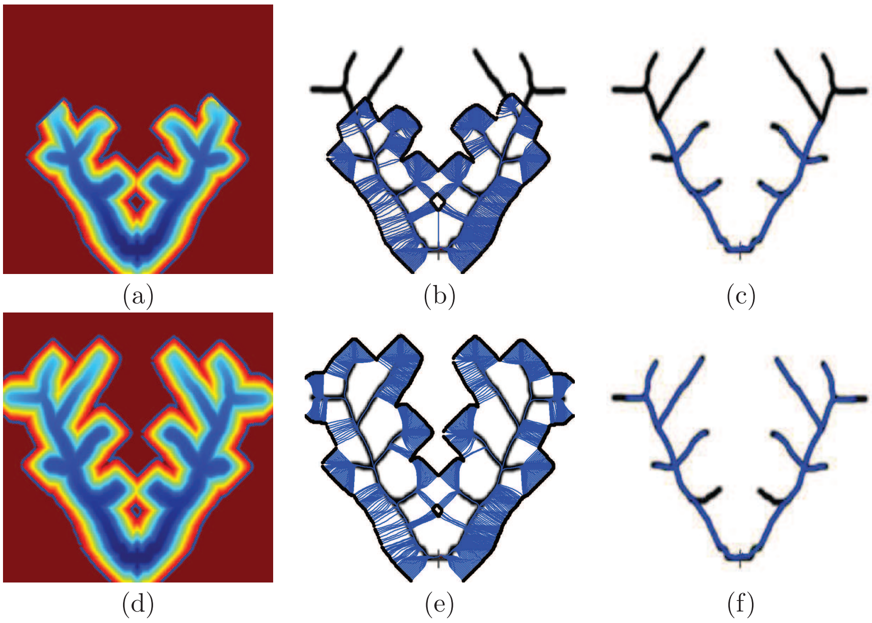


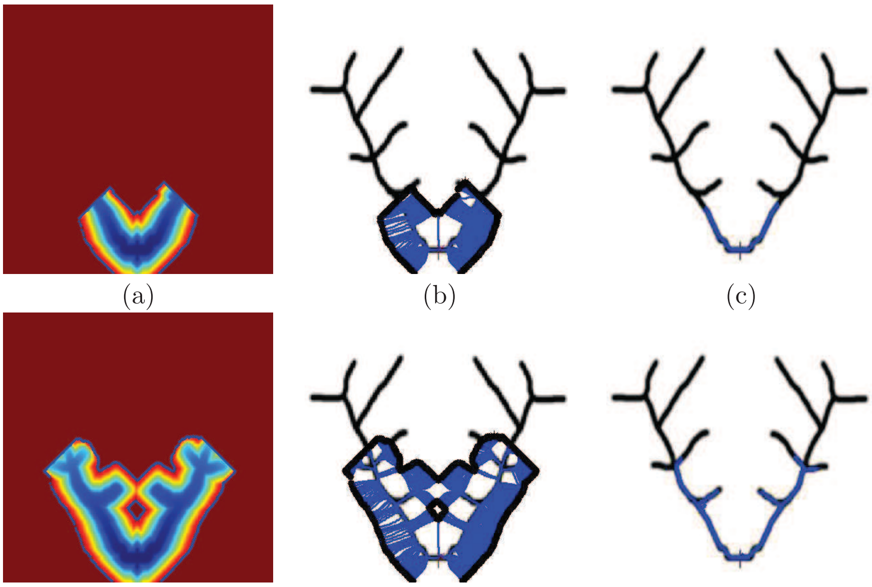
Fig. 12 Experiment on a tree structure by voting from front, from left to right: the first column shows the distance maps, the second one shows the paths obtained by backtracking from the heat front to the source point, and the third one presents the result by voting.

Voting directly from the heat front is easy to implement. When we compare the use of heat front to get end points with the use of boundary points of the image or randomly chosen end points as the end points, we find that the heat front is more suited to get the centerline. This is because the way heat diffuses depends a lot on the geometry of the shapes of the structure. Yet, as heat propagates, the heat front does not provide the exact shape of the structure to be extracted. This makes it easy to mix-up two paths that make a small angle between each other like in the case of tree structures, making it more difficult to recover all the paths. As shown in Fig.12(f), there are two segments inside the tree structure that are missing, whereas in (c), these same segments are extracted. For this reason, we propose two other methods for more complicated scenes.

4.1.2 Multiple Voting Method

As mentioned above, given a more complicated scene, such as a tree structure, there will be branches missing even if we use the boundary of the region of heat diffusion. Besides, the diffusion time ΔT should be re-selected when the image size is different. For these reasons we propose the idea of the multi-voting method.

First, a source point p_0 is given. Within a smaller period of time ΔT_1, we see a reduction in the magnitude of the region pervaded by the heat as



(a) (b) (c)

(d) (e) (f)

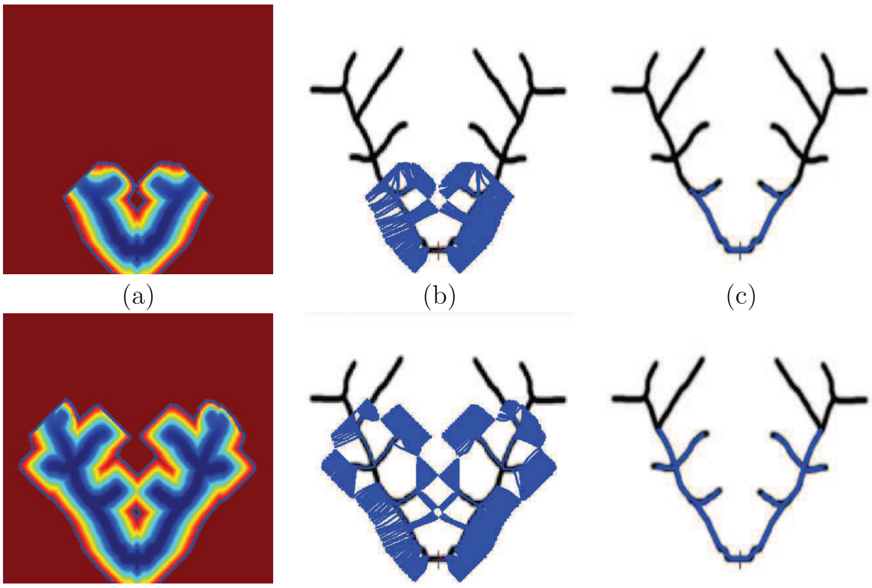
Fig. 13 Experiment on the tree structure by multiple voting method, the top row is obtained by the first step of Sect.4.1.2, the bottom row is the second step, from left to right are the distance map, the paths obtained by backtracking from the front to the source points and the result after voting.

shown in Fig.13(a). Using the boundary of this region as the end points and backtracking to p_0, as shown in (b), we vote and set a threshold ϵ of the voting score Eq.(23). Here we choose $\epsilon = \max(\beta)/7$ in the experiments. Note that we use 7 rather than 30 here because it is different from voting directly: there are fewer paths extracted at each iteration. The points $\{p_{s_1}\}$ with a higher density value than ϵ are retained, as shown in (c). Then, we let the heat continue to diffuse for the same period of time ΔT_1 and get the distance map (d). We vote again from the points on the new front of the region of heat and the newly obtained points $\{p_{s_2}\}$ are saved again as shown in (f). Heat diffuses in this way and we keep all the points $\{p_{s_1}\}$ to $\{p_{s_n}\}$ together until the stopping criterion is met (see section 4.1.4). We show the two first steps here and the final result is shown in the experiment section.

4.1.3 Accumulation of Source Points

Resetting source points method is based on a procedure similar to that of adding voting points of the previous subsection. Both of them need a multi-voting process. The resetting source points method takes advantage of diffusion and makes the process more exact because each time there will be more source points that cover the structure.

The first step of voting is the same as the one in Sect.4.1.2. After time ΔT_1 of diffusion from source point p_0, we get the points $\{p_{v1}\}$ by thresholding



(a) (b) (c)

(d) (e) (f)

Fig. 14 Experiment on the tree structure by accumulation of source points, the above row is obtained by the first step of Sect.4.1.3, the below row is the second step, from left to right are the distance map, the paths obtained by backtracking from the front to the source points and the result after voting.

the geodesic density with ϵ and save the points $\{p_{v1}\}$. We then reset the temperature to zero everywhere on the image and set all points of p_{v1} as the source points that have the same temperature. Next, we let these source points diffuse for the same time ΔT_1 and we vote again and get points $\{p_{v2}\}$. Add $\{p_{v2}\}$ to $\{p_{v1}\}$ and repeat resetting the temperature to zero on the whole domain and adding new points to $\{p_{v1}\}$ as the source points, until the stopping criterion is met. The stopping criterion here is the same as in 4.1.2 and will be detailed in sect.4.1.4.

Resetting source points method is different from Sect.4.1.2 because the source points keep changing every time there is a vote. It is better at controlling the direction of the heat flow because increasing the number of source points leads to higher accuracy. As is shown in Fig.14, these are two first intermediate steps of this algorithm. After time ΔT_1, the distance map (a) is obtained, by voting we can get (c). In the second step, we use the points in (c) as the source points, and let the heat diffuse from scratch. After the same time ΔT_1, we get the result in (f). In the next steps, we use the points obtained from its previous step as new source points until the stopping criterion is met. Compared to Fig.12, by using resetting the source points method, the segments on the tree structure are almost all extracted, and heat is more concentrated on the boundary of the region that it covers. The final result is shown in the experiment part.

4.1.4 Stopping Criterion

We define two criteria for the heat to stop diffusing. Once at least one of them is satisfied, the heat diffusion stops, see Algorithm 1. Take Fig.14 as an example.

1. At the beginning, one point p_0 is chosen as the source point in an image (with a size $M \times N$). Save p_0 into a list L, which is initialized as empty. After one step of voting, we get the points $\{p_{v1}\}$, as shown in Fig.14 (c), save them into L. Denote the number of pixels in L by N_L. If $N_L/(M \times N) > \eta$, the heat diffusion is terminated. Otherwise, it continues to diffuse. In our experiments, according to our experience, there is a rule of thumb and this magnitude η is set to 1/30.

2. After the ith step of diffusion, new points $\{p_{vi}\}$ should be added into list L, if $N_L/(N_L + N_{\{p_{vi}\}}) > 95\%$, where $N_{\{p_{vi}\}}$ is the number of the set $\{p_{vi}\}$, the heat diffusion can also be stopped.

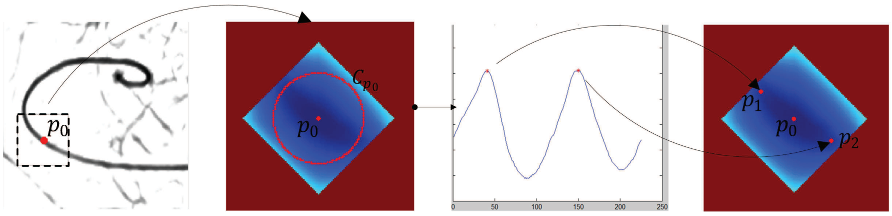
Algorithm 1 Stopping Criteria

Initialization: p_0 chosen as the source point, $L \leftarrow p_0$
repeat
    $L \leftarrow p_{v_n}$, p_{v_n}: the n-th voting;
until $N_L/(M \times N) > \eta$ or $N_L/(N_L + N_{\{p_{v_i}\}}) > 95\%$

4.2 Key Points from Heat

In [2], the authors introduced a method for segmentation using Fast Marching Method by growing minimal path and detecting key points on the curves of interest recursively. First, the user provides an initial point on the desired object. Then, starting from the initial point, a front is propagated and the key points are detected iteratively. These key points are almost equi-distributed along the curve of interest, and thus are detected based on the Euclidean lengths of the minimal paths. The whole process can be described as follows. First, start from at least one single point p_0 to initiate the Fast Marching Method. Every time we compute the geodesic distance $U_{p_0}(p_x)$ from a point p_0 to p_x on the image, we also need to compute the Euclidean length $L_{p_0}(p_x)$ of the geodesic path from p_0 to p_x. When a point p_1 satisfies that $L_{p_1} \geq \gamma$, where γ is a threshold given by the user, p_1 is considered as the first so-called key point. As soon as p_1 is detected, it is considered as a new source of propagation. The same process is used to define the successive key points $\{p_k\}$. Front propagation is let to continue until a stopping criterion is met. Refer [2] for details.

Since the key point method is efficient and robust in [2], we adapt it to heat diffusion. However, using the heat diffusion, it is not as easy to compute the



(a) (b) (c) (d)

Fig. 15 Flowchart of how to choose the new source points in the first Δt. From left to right, (a) is the original image with a source point p_0, (b) is the distance map D_1 of the dashed box in (a) after time t, the red circle is C_{p_0}, (c) plots the heat density on the points of C_{p_0} and there are two peaks which represent the new source points p_1 and p_2 in (d).

Euclidean distance together with geodesic distance as it is in the case of Fast Marching in [2,16]. This is because when using Fast Marching, the geodesic and Euclidean distances are updated simultaneously at every iteration when the status of a pixel is updated, by propagation, while in heat diffusion, we do not compute the distances by the same kind of front propagation. Therefore, we propose a new method to detect key points using heat diffusion without computing the Euclidean length of each minimal path during every detection.

First, we set a source point p_0 on the curve of interest, and p_0 can be considered as the first key point. For every key point, there are two states, $s(p_x) = 0$ and $s(p_x) = 1$, where $s(\cdot)$ is the state function. As will be made clear below, points with state 0 are used to compute the geodesic distance and lines, as well as finding the next key point, while points with state 1 are points that have already been used for computing its neighbor key points. Now let the heat diffuse from p_0, where $s(p_0) = 0$. We stop the heat from diffusing after a certain time Δt. The distance map d_{p_0} can be obtained by Eq.(3). In the region R_{p_0} of heat diffusion, we make a circle C_{p_0} where p_0 is the center and r is the radius. It should be guaranteed that the circle is located within this region R_{p_0}. Then we find the peaks among heat density $u_{C_{p_0}}$ on the circle C_{p_0} and set a threshold ϵ_1, save the positions $\{p_{s1}\}$ of the peaks whose values are larger than ϵ_1, and define $\{p_{s1}\}$ as the new source points. Fig.15 depicts the flowchart of choosing the new source point in the first Δt. We try to find the position of the peaks of the heat density. Two points p_1 and p_2 are found here, $\{p_{s1}\} = \{p_1, p_2\}$, and they are considered as the key points which are found at the first diffusion.

After the first time of diffusion Δt, we get $\{p_{s1}\}$ which denotes the new collection of source points. Since p_0 has been already computed for finding its neighbor key points, its state is changed to 1, and we use a list L to save p_0. Now it is turn for p_1 to diffuse and get its neighboring key points. We empty the heat density everywhere on the image and let heat diffuse from p_1, and use the same technique as shown in Fig.15. We get a distance map d_{p_2}. Yet, note that here we only use the subpart where $C_{p_1} \subset R_{p_1} \backslash R_{p_0}$. After finding its neighboring key point p_3, the state of p_1 is changed to 1, and p_1 is saved in the collection L. Point p_2 goes the same way until we get to p_n, where the maximal value of heat of $C_{p_n} \subset R_{p_n} \backslash \{R_{p_0}, ..., R_{p_{n-1}}\}$ is less than the threshold ε_1 we set. Algorithm.2 is the key point algorithm using the isotropic heat diffusion, where ϵ_1, ϵ_2 and ε_1 are three parameters. Once we stop the process, we have a set of key points and we obtain a set of paths, where each keypoint is linked by a geodesic path to a previously obtained key point. This is made by backtracking from a keypoint to the previous key point from which it takes its origin. Each path is obtained in the algorithm at the time the key point is added in the list.

Algorithm 2 Keypoint Algorithm

```
Initialization:
s(p_0) = 0, L ← p_0, u = 0, φ = 0, Γ = {0}; % Γ is the set of geodesic paths
repeat:
    p_curr = SearchState(L), % find the first point whose state is 0;
    u(p_curr) = 1; % initialize the heat of current point
    [u, φ] = HeatDiff(u);
    {p_m} = GetPeaks(C_{p_curr}(r)); % C_{p_curr} ⊂ R_{p_curr},
    s(p_curr) = 1;
    if max{l(p_m)} < ϵ_1
        break;
    end if
    for i = 1:m; % m is number of peak points;
        if α(p_i) > ϵ_2; % α is the conductivity in Eq.(6).
            L(end + 1) = p_i;
            Γ(end + 1) = γ(p_i, p_{i-1});
            s(L(end)) = 0;
        end if
    end for
until: max(C_{p_n} ⊂ R_{p_n} \ {R_{p_0}, ..., R_{p_{n-1}}}) < ε_1
```

5 Experiments and Analysis on Automatic Segmentation of Geodesic Curves

5.1 Experiment Data and Settings

We test on three images: 1) a synthetic tree structure image in Fig.16 and Fig.20; 2) a real vessel image with many branches in Fig.17, it is obtained by maximum intensity projection of a real 3D vessel data and 3) a real medical image with a catheter which is highly curved in Fig.19 (a), the size of all three images is 300×300. In Fig.16(b), (c), (d), we use isotropic heat equation Eq.(1) with the conductivity in all the three methods, Eq.(6), $n = 3$ here. In

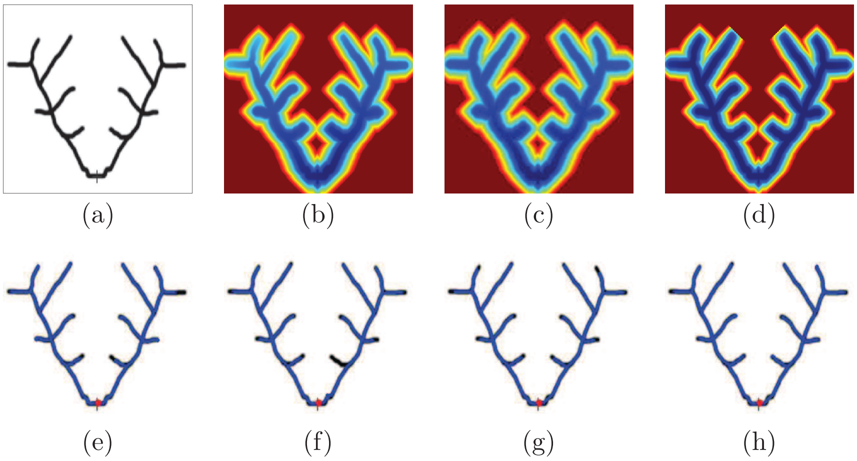


Fig. 16 Voting experiment on the tree structure, (a) is the original image, (e) is the result by [29] by using randomly chosen end points, from the second column to the fourth, the above row shows the voting map by using respectively voting from the front, multi-voting by adding voting points and multi-voting by accumulating of source points. And on the bottom row are the corresponding results extracted by the three automatic methods.

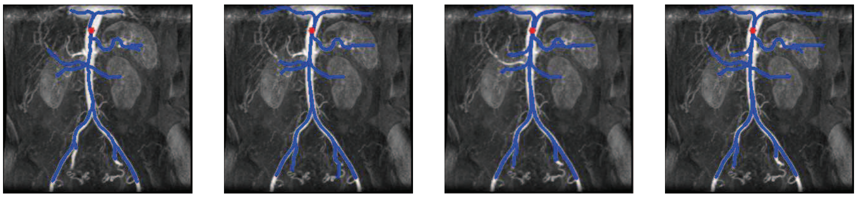
Fig.17 , Fig.18 and Fig.19, we use Eq.(10), the combination of P-M model and conductivity for the heat diffusion. In the process of voting directly from the front, the time ΔT is controlled by the iteration times of heat diffusion, we set the iteration times to $2N$, where N is the length of image, here $N = 300$. In the process of multiple voting methods, including Sect.4.1.2 and Sect.4.1.3, the smaller time ΔT_1 is controlled by the iteration time of each step, which is set to 100 here. In detecting key points, for each step, the iteration time is 50, and the radius r is 30, the threshold ϵ_1 we set here is 80% of the highest heat density on C_{p_0} during the first iteration.

5.2 Results and Analysis

The above results in Sect.3 show the semi-automatic extraction of geodesic lines by setting the source and end points manually. Here we extract the centerlines by voting method or using the key point method.

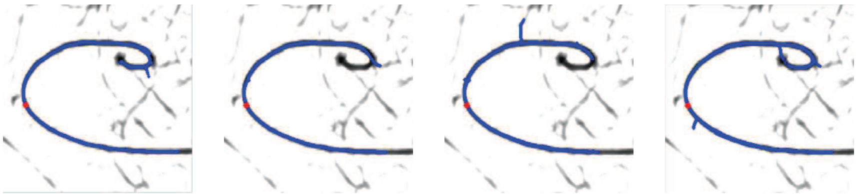
In Fig.16, (e) is the result obtained by [29] and 500 end points are randomly chosen on the image. (b), (c) and (d) are the voting maps obtained by using directly voting from the front, multi-voting by adding voting points and multi-voting by accumulating of source points. (f), (g) and (h) show the corresponding structures that are extracted by different methods, and the black part are mis-extracted part. In (e), there are some parts missing in the terminals in the tree branches, the same phenomenon takes place in (f) and (g). (h) has the best result because all branches and details, as well, are extracted.

Fig.17 shows the results of Fast Marching (a) and the three automatic voting methods (b) (c) (d) with heat on real medical images. From the results,



(a) (b) (c) (d)

Fig. 17 Voting experiment on the real vessel image, (a) is the original image with classical voting from Fast Marching, (b) (c) (d) are the results by using directly voting from the front, multi-voting by adding voting points and multi-voting by accumulating of source points.



(a) (b) (c) (d)

Fig. 18 Voting experiment on the real medical image with catheter, from left to right: the results of [29] are displayed by using randomly chosen end points; using directly voting from the front; multi-voting by adding voting points and multi-voting by accumulating of source points.

we can see that the blue lines in (b) (c) (d) go along the centerline of the vessels, but (a) fails to follow the centerline. And for all of (a), (b) and (c), there are segments missing. However, by using resetting source point method nearly all vessels are extracted in (d).

In Fig.18, the terminals of the catheter are hard to extract. Results are shown superimposed on the potential image built from the Laplacian. Fig.18(a) is the result by [29] and the end points are randomly selected, and from (b) through (d) are the results from using directly voting from the front, multi-voting by adding voting points, and multi-voting by accumulation of source points. All of the four results give the contour of the curve-of-interest.

The automatic method in Sect.4.2 is tested in Fig.19 and Fig.20. Fig.19(a) is the original medical image, where there is a curve with some high curvature; (b) is the potential built according to the Laplacian; (c) is the first step through detecting key points, and we can see that there are two points (yellow) that are detected; (d) is the step following (c), and it can be found that there is another key point detected; (e) is the next step and it is also the third step in the whole process of detection; (f) is the result after several steps and (g) is the final result in the case where the blue paths that are extracted by backtracking from each key point one after the other, and they cover exactly the curve; (h) is the traveling map of the key points, which means the set of the geodesic distance map of each key point of the whole process of heat diffusion.

The key point from heat method does better than voting methods for this example. Computationally, comparing the heat density becomes much easier

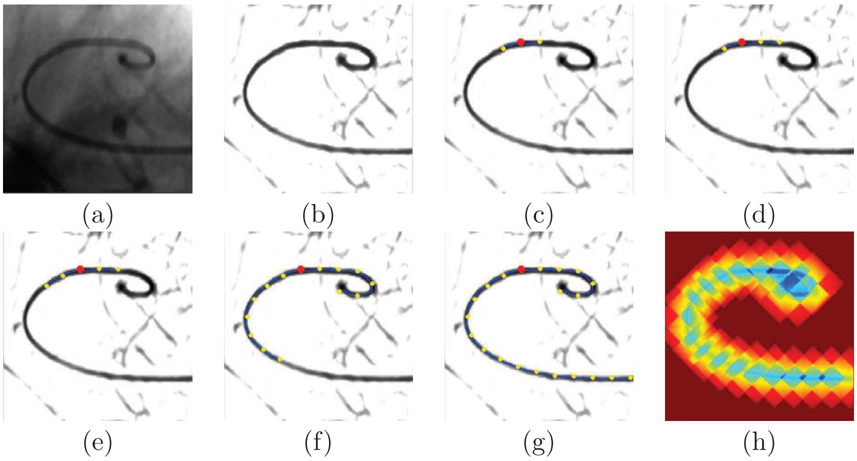


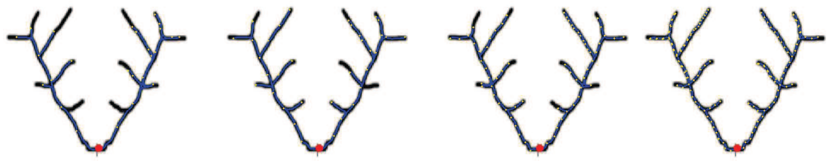
Fig. 19 Keypoint experiment: (a) is the original image; (b) is the Laplacian of the original image; (c) the red point is the source point given by user, and the two yellow points are the first key points detected; (d) is the second step of detecting key point from heat; (e) is the next step, and (f) the result after several steps in search for key points; (g) shows the final result of all the key points and paths detected and (h) is the traveling map of key points.

and less time-consuming than computing the Euclidean length of the geodesic curves each time in [2]. Furthermore, it also provides satisfactory results.

Another important issue in using key points from heat is how to choose an appropriate radius r for defining the circles every time. Fig.20 are experiments on the tree structure using the key points from heat. From (a) through (d), the radius are 45, 30, 15, 10 respectively. And from the extracted paths, we can see that (d) gives the most complete structure among the four results. In this case, it can be seen that when a smaller radius r is used, the result is better. But this does not prove that it is better using a smaller r in every situation. In fact, how to choose an appropriate radius depends on the size of the feature that we want to extract. Here in the tree structure, the width of the branches of the tree have no more than 10 pixels, so a smaller radius is more effective in key points detection in this particular case. However, as in the classical key point method [2], using a larger radius makes the method less sensitive to noise, and this is why the radius should not be chosen too small.

Acknowledgements

We would like to thank the reviewers, as well as Aristide-Oswald Bartet, for their useful comments that allowed us to improve this paper. Special thanks to Dr.Vivek Kaul and Prof.Anthony Yezzi who made a complete reading of our paper in order to check for correct English and helped for this revision. Many thanks to Jean-Marie Mirebeau, Dario Prandi and Gabriel Peyré for fruitful discussions.



(a) (b) (c) (d)

Fig. 20 Keypoint experiment on tree structure by using key point method from heat, the red point is the source point given by the user, and the yellow points are the key points detected. From left to right are displayed the results when using different radii r in defining the circles, they are 45, 30, 15 and 10 pixels respectively from (a) to (d).

6 Conclusion

We proposed new methods using the isotropic and anisotropic heat diffusions to get the geodesic distance and geodesic lines for image segmentation purposes. Using different kinds of diffusivity, models and tensors, the methods work well for different types of images and features of interest. For example, the P-M model can either try to hold the heat within the boundary of a region or make the heat flow along the edges. By using different diffusion tensors, the anisotropic heat diffusion will flow along the direction that we design. The biggest advantage of using heat flow is that it is very fast and robust as well, and also easy to implement. Furthermore, heat diffusion is not very sensitive to noise, a little noise will not affect its performance.

In addition, this paper introduces two new automatic methods, based on extension to heat method of geodesic voting and detecting key points for the extraction of geodesic centerlines. Experiments show that using the proposed automatic methods, the results are satisfactory and robust as well as time-saving.

In future, we will try to find more appropriate forms of conductivity and tensors for heat diffusion to get the centerlines as well as the boundary of the vessels. We will also extend heat methods for automatic segmentation of vessels in 3D medical images.

References

1. Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 485–493. ACM, 2003.
2. Fethallah Benmansour and Laurent D. Cohen. Fast object segmentation by growing minimal paths from a single point on 2D or 3D images. *Journal of Mathematical Imaging and Vision*, 33(2):209–221, 2009.
3. Fethallah Benmansour and Laurent D. Cohen. Tubular anisotropy segmentation. In *Scale Space and Variational Methods in Computer Vision*, pages 14–25. Springer, 2009.
4. Laurent D. Cohen and Ron Kimmel. Global minimum for active contour models: A minimal path approach. *International journal of computer vision*, 24(1):57–78, 1997.

5. David Cohen-Steiner and Jean-Marie Morvan. Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 312–321. ACM, 2003.
6. Radu Constantinescu, Nick Costanzino, Anna L. Mazzucato, and Victor Nistor. Approximate solutions to second order parabolic equations i: analytic estimates. *arXiv preprint arXiv:0910.1562*, 2009.
7. Richard Courant, Kurt Friedrichs, and Hans Lewy. On the partial difference equations of mathematical physics. *IBM journal of Research and Development*, 11(2):215–234, 1967.
8. Keenan Crane, Clarisse Weischedel, and Max Wardetzky. Geodesics in heat: a new approach to computing distance based on heat flow. *ACM Transactions on Graphics (TOG)*, 32(5):152, 2013.
9. Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.
10. Jérôme Fehrenbach and Jean-Marie Mirebeau. Sparse non-negative stencils for anisotropic diffusion. *Journal of Mathematical Imaging and Vision*, 49(1):123–147, 2014.
11. Alejandro F. Frangi, Wiro J. Niessen, Koen L. Vincken, and Max A. Viergever. Multiscale vessel enhancement filtering. In *Medical Image Computing and Computer-Assisted Intervention, MICCAI98*, pages 130–137. Springer, 1998.
12. Jiuxiang Hu, Anshuman Razdan, John C. Femiani, Ming Cui, and Peter Wonka. Road network extraction and intersection detection from aerial images by tracking road footprints. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(12):4144–4157, 2007.
13. Saâd Jbabdi, Pierre Bellec, Roberto Toro, Jean Daunizeau, Mélanie Pélégrini-Issac, and Habib Benali. Accurate anisotropic fast marching for diffusion-based geodesic tractography. *Journal of Biomedical Imaging*, 2008:2, 2008.
14. J.Sun, M.Ovsjanikov, and L.Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum*, 28(5):1383–1392, 2009.
15. J.Weickert. *Anisotropic diffusion in image processing*. Teubner-Verlag, 1998.
16. Vivek Kaul, Anthony Yezzi, and Yichang Tsai. Detecting curves with unknown endpoints and arbitrary topology using minimal paths. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(10):1952–1965, 2012.
17. Sermesant M. Clatz O. Peyrat J. M. Delingette H. Konukoglu, E. and N. Ayache. A recursive anisotropic fast marching approach to reaction diffusion equation: Application to tumor growth modeling. In *Information processing in medical imaging*, 2007.
18. Ullrich Köthe. Edge and junction detection with an improved structure tensor. In *Pattern Recognition*, pages 25–32. Springer, 2003.
19. Max WK Law and Albert CS Chung. Three dimensional curvilinear structure detection using optimally oriented flux. In *Computer Vision–ECCV 2008*, pages 368–382. Springer, 2008.
20. Jean-Marie Mirebeau. Anisotropic fast-marching on cartesian grids using lattice basis reduction. *SIAM Journal on Numerical Analysis*, 52(4):1573–1599, 2014.
21. Jean-Marie Mirebeau, Jérôme Fehrenbach, Laurent Risser, and Shaza Tobji. Anisotropic diffusion in ITK. *arXiv preprint arXiv:1503.00992*, 2015.
22. E Pardoux. Stochastic partial differential equations and filtering of diffusion processes. *Stochastics*, 3(1-4):127–167, 1980.
23. Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(7):629–639, 1990.
24. Gabriel Peyré, Mickael Péchaud, Renaud Keriven, and Laurent D. Cohen. Geodesic methods in computer vision and graphics. *Foundations and Trends® in Computer Graphics and Vision*, 5(3–4):197–397, 2010.
25. Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15–36, 1993.
26. Dan Raviv and Ron Kimmel. Affine invariant geometry for non-rigid shapes. *International Journal of Computer Vision*, 111(1):1–11, 2015.

27. Martin Reuter, Silvia Biasotti, Daniela Giorgi, Giuseppe Patanè, and Michela Spagnuolo. Discrete laplace–beltrami operators for shape analysis and segmentation. *Computers & Graphics*, 33(3):381–390, 2009.
28. Youssef Rouchdy and Laurent D. Cohen. Image segmentation by geodesic voting. application to the extraction of tree structures from confocal microscope images. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–5. IEEE, 2008.
29. Youssef Rouchdy and Laurent D. Cohen. A geodesic voting method for the segmentation of tubular tree and centerlines. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pages 979–983. IEEE, 2011.
30. Youssef Rouchdy and Laurent D. Cohen. Geodesic voting for the automatic extraction of tree structures. *Computer Vision and Image Understanding*, 117(10):1453–1467, 2013.
31. James A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
32. James A. Sethian and Alexander Vladimirsky. Fast methods for the eikonal and related hamilton–jacobi equations on unstructured meshes. *Proceedings of the National Academy of Sciences*, 97(11):5699–5703, 2000.
33. S.R.S.Varadhan. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics*, 20(2):431–455, 1967.
34. Thomas JS Taylor. Off diagonal asymptotics of hypoelliptic diffusion equations and singular riemannian geometry. *Pacific journal of mathematics*, 136(2):379–399, 1989.
35. J Unsworth and FJ Duarte. Heat diffusion in a solid sphere and fourier theory: An elementary practical example. *American Journal of Physics*, 47(11):981–983, 1979.
36. Dmitri V. Vassilevich. Heat kernel expansion: user's manual. *Physics reports*, 388(5):279–360, 2003.
37. Joachim Weickert. Coherence-enhancing diffusion filtering. *International Journal of Computer Vision*, 31(2-3):111–127, 1999.
38. Andrew P. Witkin. Scale-space filtering: A new approach to multi-scale description. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'84.*, volume 9, pages 150–153. IEEE, 1984.