A Second Order Multi-Stencil Fast Marching Method with a Non-Constant Local Cost Model

Susana Merino-Caviedes, Lucilio Cordero-Grande, María Teresa Pérez, Pablo Casaseca-de-la-Higuera, Marcos Martín-Fernández, Rachid Deriche and Carlos Alberola-López

Abstract—The Fast Marching method is widely employed in several fields of image processing. Some years ago a Multi-Stencil version (MSFM) was introduced to improve its accuracy by solving the equation for a set of stencils and choosing the best solution at each considered node. The following work proposes a modified numerical scheme for MSFM to take into account the variation of the local cost, which has proven to be second order. The influence of the stencil set choice on the algorithm outcome with respect to stencil orthogonality and axis swapping is also explored, where stencils are taken from neighborhoods of varying radius. The experimental results show that the proposed schemes improve the accuracy of their original counterparts, and that the use of permutation-invariant stencil sets provides robustness against shifted vector coordinates in the stencil set.

Index Terms—Differential equations, Eikonal equation, Fast marching methods, finite differences, multi-stencil schemes.

I. INTRODUCTION

The Fast Marching (FM) method was proposed in [1] in the context of propagating fronts, and it has since been widely employed for several applications, such as the extraction of curve skeletons from 3D shapes [2], [3], shape from shading [4], [5], image inpainting [6] and others (see [7]). In the context of image processing, the FM is a computationally efficient means to approximate Euclidean or weighted distance maps. For example, it has been widely employed in the active contour framework to generate initial level set functions and/or to provide a rough approximation to the solution [8], [9]. In addition, modifications of the FM method have been proposed for other applications such as the computation of brain connectivity

S. Merino-Caviedes, P. Casaseca-de-la-Higuera, M. Martín-Fernández and C. Alberola-López are with the Laboratorio de Procesado de Imagen, Universidad de Valladolid, Valladolid, Spain (contact email: smercav@lpi.tel.uva.es).

L. Cordero-Grande is with the Centre for the Developing Brain and Department of Biomedical Engineering, Division of Imaging Sciences and Biomedical Engineering, King's College London, London, UK.

M. T. Pérez is with the Dpto. de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain.

R. Deriche is with the Athena Project-Team, INRIA Sophia-Antipolis Méditerranée, Sophia Antipolis, France.

maps [10], blood flow [11] and myocardial scar transmurality, which has clinical relevance in the diagnosis and prognosis of cardiomyopathies [12].

Several schemes to improve the accuracy and computational efficiency of the FM method have been proposed. One of the earliest works suggested a second-order finite difference scheme for the approximation of the derivatives [7]. The Multi-Stencil Fast Marching (MSFM) method [13] reformulates the Eikonal equation in terms of directional derivatives, and this makes it possible to use different stencils. The MSFM allows for the use of second-order approximations for the derivatives, and also improves the FM accuracy along the diagonal directions represented by the stencil vectors. The local cost in the original FM method and in many of the suggested numerical schemes for improving its accuracy, such as MSFM, is considered constant in the neighborhood.

In [14], the problem of finding globally optimal trajectories, given a local cost and a starting location, is solved using the Hopf-Lax formulation of Hamilton-Jacobi equations. The function value at each node is computed by means of a minimization problem. Some works following this methodology have addressed the local cost issue. The Shifted Grid Fast Marching (SGFM) method [15] proposed shifting the local cost sampling grid with respect to the image grid, and use those values in the node update problem. In [16], it was shown that the estimated global cost of traveling from a point A of the image to a point B was not necessarily equal to that of traveling from B to A, due to anisotropies in the discrete algorithm, a problem they solved by employing a linear model for the local cost and several interpolation-derived schemes. The methods [15] and [16], however, do not allow for the construction of a second-order scheme, and its application to higher dimensional problems is not straightforward.

In this paper, a modification of the MSFM numerical scheme is proposed where, as in [16], a linear model is used for the local cost but, unlike [16], it is second-order accurate. This is achieved by evaluating the local cost at the center of the parallelepiped generated by the considered stencil vectors and using a second order finite difference scheme employing the parallelepiped vertices for the derivatives. In addition, the scheme formulation may also be used for regular anisotropic grids and non orthogonal stencils. Experimental results show that, in most cases, the proposed method improves the accuracy with respect to the original method. The influence of the choice of stencil sets on the results is also explored. We show that the error produced by the 3D 6-stencil set employed in the original MSFM method is not invariant to dimension

permutations of the stencil set vectors for functions with radial symmetry. To avoid this, the use of permutation invariant stencil sets is preferable. In addition, the results of orthogonal or nearly-orthogonal stencils in neighborhoods of increasing radius are more accurate than those of a set of highly non-orthogonal stencils.

The proposed MSFM method is applied to optimal course planning of an unmanned aerial vehicle (UAV) in a flooding episode; specifically, the purpose is to obtain a trade-off between the distance travelled by the UAV and the amount of visited flooded land so as to permit close inspections. We show that our MSFM method outperforms the previously described alternative FM methods according to different criteria related with the pursued trade-off.

The rest of the document is structured as follows. In Section II, the original MSFM method is described. Then, in Section III, the variable local cost scheme is proposed for MSFM, along with the definition of permutation invariant stencil sets. Section IV contains the experimental results obtained with analytical functions with respect to accuracy, convergence and stencil permutations. Performance evaluation on the UAV path planning case study above described is presented in Section V. Our conclusions are summarized in Section VI. Finally, the proof that the proposed scheme is second order is given in an Appendix.

II. MULTI-STENCIL FAST MARCHING METHOD

A. Description

For the sake of completeness, this section provides a brief overview of the original FM method [1] and the MSFM method [13]. For an in-depth description of the aforementioned methods, please refer to the original materials. The original FM method was proposed to solve in one pass the boundary problem:

$$|\nabla T(\mathbf{x})| \, F(\mathbf{x}) = 1, \text{s.t. } T(\Gamma_0) = 0 \qquad (1)$$

defined in $\Omega \subset \mathbb{R}^D$, where $\mathbf{x} \in \Omega$ and $T(\mathbf{x})$ is the arrival time of a front whose initial location is given by $\Gamma_0 \subset \Omega$ and moves along the normal direction with local speed $F(\mathbf{x}) > 0$. In this work, the problem dimensionality is restricted to $D \in \{2, 3\}$. Occasionally, Eq. (1) is instead expressed in terms of the local cost $C(\mathbf{x}) = 1/F(\mathbf{x})$. Taking advantage that $T(\mathbf{x})$ is the arrival time of a monotonically advancing front, and following the philosophy of Dijkstra's shortest path algorithm [17], the numerical solution is computed in one pass by propagating the solution from the node with the smallest $T(\mathbf{x})$ to its neighbors, starting from Γ_0. This is done by means of a finite-difference upwind scheme, where the derivative approximations are computed along the image axes; that is, only a single stencil is employed.

The MSFM method, proposed in [13], adds the information of nodes not aligned with the image axes in the computation of $T(\mathbf{x})$. Like the FM algorithm, it solves the problem in Eq. (1) in one pass, and its general algorithm is given in Fig. 1. Let $\mathcal{Q} = \left\{\mathcal{S}^i\right\}_{i=1}^Q$ be a set of Q stencils $\mathcal{S}^i = \left\{\mathbf{v}_j^i\right\}_{j=1}^D$, each of which constitutes a basis for $\mathbb{R}^D$. The MSFM reformulates Eq. (1) so that it is expressed

Require: Γ_0, $\mathcal{Q}$ and $F(\mathbf{x}), \forall \mathbf{x} \in \Omega$.
  Output $T(\Omega \setminus \Gamma_0) \leftarrow \infty$, $T(\Gamma_0) \leftarrow 0$
  Labels $L(\Omega \setminus \Gamma_0) \leftarrow Far$, $L(\Gamma_0) \leftarrow Frozen$
  for all $\mathbf{x}_n \leftarrow Far$ neighbors of Γ_0 **do**
      for all $\mathcal{S}^i \in \mathcal{Q}$ **do**
          Compute $T^i(\mathbf{x}_n)$ with $\mathcal{S}^i$, $T(\cdot)$ and $F(\cdot)$
      end for
      $T(\mathbf{x}_n) \leftarrow \min(\{T^i(\mathbf{x}_n)\}_{i=1}^Q)$
      $L(\mathbf{x}_n) \leftarrow Active$
  end for
  while Any $Active$ node exists **do**
      $\mathbf{x}_f \leftarrow \underset{\mathbf{x}}{\arg\min}\, T(\mathbf{x})$, s.t. $L(\mathbf{x}) = Active$
      $L(\mathbf{x}_f) \leftarrow Frozen$
      for all $\mathbf{x}_n \leftarrow Far$ or $Active$ neighbors of $\mathbf{x}_f$ **do**
          for all $\mathcal{S}^i \in \mathcal{Q}$ **do**
              Compute $T^i(\mathbf{x}_n)$ with $\mathcal{S}^i$, $T(\cdot)$ and $F(\cdot)$
          end for
          $T(\mathbf{x}_n) \leftarrow \min(\{T^i(\mathbf{x}_n)\}_{i=1}^Q)$
          if $L(\mathbf{x}_n) = Far$ **then**
              $L(\mathbf{x}_n) \leftarrow Active$
          end if
      end for
  end while

Figure 1. MSFM general algorithm.

on directional derivatives of $T(\mathbf{x})$ instead of $\nabla T(\mathbf{x})$. Let $\psi_j^i(\mathbf{x}) = \langle \nabla T(\mathbf{x}), (\mathbf{v}_j^i/\|\mathbf{v}_j^i\|) \rangle$ be the directional derivative of $T(\mathbf{x})$ along $\mathbf{v}_j^i$ and $\Psi^i(\mathbf{x}) = (\psi_1^i, \ldots, \psi_D^i)^T$. Notice that $\Psi^i(\mathbf{x})$ and $\nabla T(\mathbf{x})$ are linearly related by $\Psi^i(\mathbf{x}) = R^i \nabla T(\mathbf{x})$, where R^i is a $D \times D$ matrix with elements $R_{jk}^i = \frac{v_{j,k}^i}{\|\mathbf{v}_j^i\|}$, and $v_{j,k}^i$ is the k-th element of $\mathbf{v}_j^i$. Equation (1) may thus be expressed as:

$$\Psi^i(\mathbf{x})^T G^i \Psi^i(\mathbf{x}) = \frac{1}{F(\mathbf{x})^2} = C(\mathbf{x})^2, \quad \text{s.t. } T(\Gamma_0) = 0 \quad (2)$$

with $G^i = (R^i(R^i)^T)^{-1}$. For the sake of simplicity, the final mathematical developments and numerical schemes provided in [13] are given for stencils with orthogonal vectors and an isotropic grid spacing, hence $G^i = Id$, where Id is the $D \times D$ identity matrix. Therefore, Eq. (2) becomes:

$$\sum_{j=1}^D \psi_j^i(\mathbf{x})^2 = \frac{1}{F(\mathbf{x})^2} = C(\mathbf{x})^2, \quad \text{s.t. } T(\Gamma_0) = 0 \quad (3)$$

Like the original FM method, Eq. (2) is numerically computed by an upwind finite difference scheme. The directional derivatives $\psi_j^i(\mathbf{x})$ are approximated by:

$$\psi_j^i(\mathbf{x}) \approx \max\left(D^{-\mathbf{v}_j^i} T(\mathbf{x}), -D^{+\mathbf{v}_j^i} T(\mathbf{x}), 0\right) \qquad (4)$$

where $D^{+\mathbf{v}_j^i} T(\mathbf{x})$ and $D^{-\mathbf{v}_j^i} T(\mathbf{x})$ are respectively the forward and backward finite difference schemes:

$$D^{\mp\mathbf{v}_j^i} T(\mathbf{x}) = \frac{T(\mathbf{x}) - T(\mathbf{x} \mp \mathbf{v}_j^i)}{\pm\|\mathbf{v}_j^i\|} \qquad (5)$$

for the first-order approximation, and:

$$D^{\mp \mathbf{v}_j^i} T(\mathbf{x}) = \frac{3T(\mathbf{x}) - 4T(\mathbf{x} \mp \mathbf{v}_j^i) + T(\mathbf{x} \mp 2\mathbf{v}_j^i)}{\pm 2\|\mathbf{v}_j^i\|} \quad (6)$$

for the second-order approximation. The requirements to use Eq. (6) are that $T(\mathbf{x} \mp \mathbf{v}_j^i) > T(\mathbf{x} \mp 2\mathbf{v}_j^i)$ and $T(\mathbf{x} \mp 2\mathbf{v}_j^i)$ is *Frozen*[1]. Plugging Eq. (4) into Eq. (3) yields a quadratic equation for the estimated $T(\mathbf{x})$ using $\mathcal{S}^i$, $T^i(\mathbf{x})$, of the form:

$$\sum_{j=1}^{D} g_{\nu_j}^i \left(T^i(\mathbf{x})^2 - 2T_{\nu_j}^i T^i(\mathbf{x}) + (T_{\nu_j}^i)^2 \right) = C(\mathbf{x})^2 \quad (7)$$

where $g_{\nu_j}^i$ is a function that depends on $\mathcal{S}^i$ and the grid spacing, $T_{\nu_j}^i = \min(T(\mathbf{x} - \mathbf{v}_j^i), T(\mathbf{x} + \mathbf{v}_j^i))$ for the first order approximation, and

$$T_{\nu_j}^i = \min \left(\frac{4T(\mathbf{x} - \mathbf{v}_j^i) - T(\mathbf{x} - 2\mathbf{v}_j^i)}{3}, \right.$$
$$\left. \frac{4T(\mathbf{x} + \mathbf{v}_j^i) - T(\mathbf{x} + 2\mathbf{v}_j^i)}{3} \right) \quad (8)$$

for the second order approximation. Only *Frozen* nodes are allowed to take part in the computation.

Out of the two roots of Eq. (7), the maximum is chosen. Nevertheless, to be accepted as a valid solution, it must fulfill the causality condition imposed by the monotonicity of the front motion, that is, $T^i(\mathbf{x}) > \max(\{T_{\nu_j}^i\}_{j=1}^D)$. While this condition is not met, Eq. (7) is recalculated using only the nodes for which $T^i(\mathbf{x}) > T_{\nu_j}^i$. The value for $T(\mathbf{x})$ is chosen as the minimum $T^i(\mathbf{x}), i = 1, \ldots, Q$ that fulfills the causality condition. Viewing $T(\mathbf{x})$ as the arrival time of a monotonically moving front, the interpretation of this criterion is that the shortest arrival time among all the viable candidates is selected for this node.

B. Limitations

The MSFM improves the accuracy of the original FM method. For 3-D functions, [13] proposes the stencil set $\mathcal{Q}_o$ given in Table I, referring to $\mathcal{S}^{o1}$ as the face-based stencil, to $\mathcal{S}^{o2}$–$\mathcal{S}^{o4}$ as edge-connected stencils, and to $\mathcal{S}^{o5}$–$\mathcal{S}^{o6}$ as vertex-connected stencils. The first limitation is that, contrary to $\mathcal{S}^{o1}$–$\mathcal{S}^{o4}$, the vertex-connected stencils $\mathcal{S}^{o5}$–$\mathcal{S}^{o6}$ are not orthogonal bases of $\mathbb{R}^3$. The use of these stencils affects the spatial error distribution of the MSFM output in that the error will not be necessarily symmetric, even for radial functions on an isotropic grid, as Fig. 2 illustrates. Figures 2a, 2b and 2c show the absolute error of the MSFM output with $F(\mathbf{x}) = 1$ using the 6-stencil set $\mathcal{S}^{o1}$–$\mathcal{S}^{o6}$ respectively for the slices in the XY, XZ and YZ planes passing through the origin of coordinates. We apply the six possible axes permutations of the computed absolute error for the $\mathcal{Q}_o$ stencil set and compare the results by pairs. Figure 2d shows the projection on the XY plane of the worst case maximum deviation between the permutation pairs along the z-axis. Observe that the maximum value of the worst case maximum deviation is about twice the maximum local error on the XY, XZ and YZ planes.

---

[1]Here, *Frozen* means that the node value will not change in the rest of the algorithm computation (see Fig. 1).

Table I
3-D Stencil Set $\mathcal{Q}_o$ proposed in the original MSFM method.

Stencil	Vectors
$\mathcal{S}^{o1}$	$(1,0,0)^T, (0,1,0)^T, (0,0,1)^T$
$\mathcal{S}^{o2}$	$(1,0,0)^T, (0,1,-1)^T, (0,1,1)^T$
$\mathcal{S}^{o3}$	$(0,1,0)^T, (-1,0,1)^T, (1,0,1)^T$
$\mathcal{S}^{o4}$	$(0,0,1)^T, (1,-1,0)^T, (1,1,0)^T$
$\mathcal{S}^{o5}$	$(1,0,1)^T, (-1,1,1)^T, (1,1,-1)^T$
$\mathcal{S}^{o6}$	$(1,0,-1)^T, (1,1,1)^T, (-1,1,-1)^T$

The second limitation is that the computation of $T(\mathbf{x})$ approximates the local speed $F(\mathbf{x})$ in the neighborhood as a constant, which is not the case in general. Consider a 1-D example with a sinusoidal local cost $C_a(x) = 1/F(x) = 1 - \cos((\pi x)/4)$. $T(x)$ may be analytically obtained, and it has the expression $T_a(x) = x - 4\sin((\pi x)/4)/\pi$ for $x \geq 0$ if $T_a(0) = 0$ is applied. For this problem, the FM computed on a 1-D grid with $h = 1$ has the expression $T_{est}(kh) = T_{est}((k-1)h) + hC_a(kh)$ for $k > 0$. Figure 3 shows $C_a(x)$, $T_a(x)$, $T_{est}(x)$, the error $T_{est}(x) - T_a(x)$, and in black the contributions of each node to $T_{est}(x)$. It may be observed how approximating $C(\mathbf{x})$ by a constant leads to a significant error. This effect is well known in numerical integration, and one of the simplest rules that solves this is the mid-point rule, which evaluates the function to be integrated at the middle point of each interval.

III. PROPOSED IMPROVEMENTS

A. Proposed Node Update

The exposition below is given for $D = 3$, although it can be easily generalized to any $D \geq 1$. Let $\mathbf{x}$ be the node where $T(\mathbf{x})$ is to be updated with a stencil $\mathcal{S}^i = \{\mathbf{v}_j^i\}_{j=1}^D$ (not necessarily orthogonal). The proposed scheme approximates the directional derivative $\psi_j^i(\cdot)$ on the center of the parallelepiped (or parallelogram, in 2D) generated by the vectors of the considered stencil $\mathcal{S}^i = \{\mathbf{v}_j^i\}_{j=1}^D$. Assume that every node in the parallelepiped, except for $\mathbf{x}$, has an already computed and *Frozen* value for $T(\cdot)$. The parallelepiped contains two faces parallel to the plane generated by $\mathcal{S}^i \setminus \{\mathbf{v}_j^i\}$: one face containing the node $\mathbf{x}$, and the other, the node $\mathbf{x} \pm \mathbf{v}_j^i$. Figures 4a and 4b illustrate this scenario, when $\mathbf{v}_j^i = \mathbf{v}_3^i$ and the *Frozen* node along $\mathbf{v}_j^i$ lies backwards and forwards, respectively, from $\mathbf{x}$. Without loss of generality, the following description focuses on the situation depicted in Fig. 4a; the derivation of the scheme when one or more *Frozen* nodes lie in the opposite direction can be easily derived from it, just by changing the signs of their respective stencil vectors. The nodes at the parallelepiped vertices in the backward approximation are divided into two groups: $\mathcal{N}_j^{0,i}$ and $\mathcal{N}_j^{-,i}$, depending on whether they belong to the face containing $\mathbf{x}$ or $\mathbf{x} - \mathbf{v}_j^i$, respectively.

The application point of the finite differences approximation to the derivatives in the proposed scheme is moved to the geometric center $\mathbf{x}_c^i$ of all the parallelepiped vertices. Consequently, the local cost is sampled at $\mathbf{x}_c^i$ as well: either $F(\mathbf{x}_c^i)$ or $C(\mathbf{x}_c^i)$ is employed. The forward and backward node values
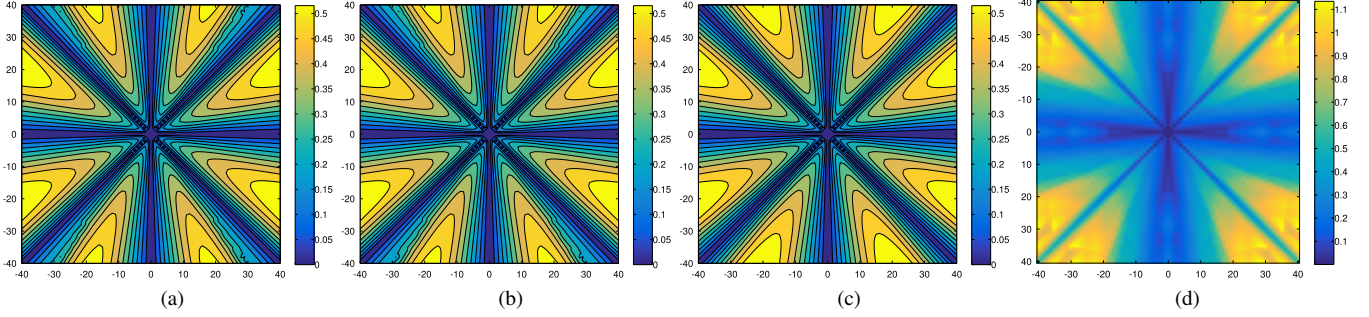
Figure 2. Local absolute error for $F(\mathbf{x}) = 1$ using MSFM on an isotropic grid. Approximation using the six stencils proposed in [13] on the planes (a) XY, (b) XZ and(c) YZ, and (d) the worst case absolute deviation along the Z axis projected onto the XY plane.
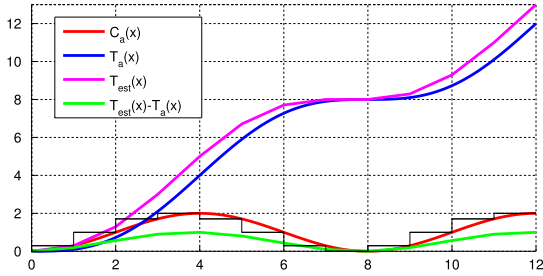

Figure 3. 1-D comparison of an analytical solution with the FM estimation and its error.
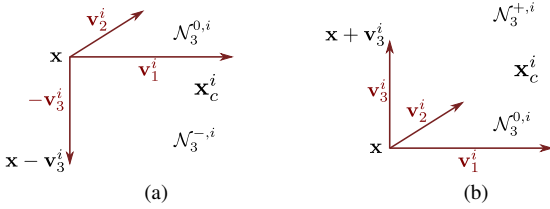

Figure 4. Node disposition for the (a) backward and (b) forward configurations for the derivative approximation along $\mathbf{v}_3^i$ in the proposed scheme.

are evaluated by averaging the $T(\cdot)$ values of $\mathcal{N}_j^{0,i}$ and $\mathcal{N}_j^{-,i}$, respectively; these forward and backward values may also be interpreted as a barycentric interpolation of $T(\mathbf{x})$ at the center of their respective parallelepiped faces (using vertices, exclusively). Thus, the proposed backward scheme $\widehat{D}^{-\mathbf{v}_j^i}T(\mathbf{x})$ for $\psi_j^i(\mathbf{x})$ is:

$$\widehat{D}^{-\mathbf{v}_j^i}T(\mathbf{x}) = \frac{\displaystyle\sum_{\hat{\mathbf{z}}\in\mathcal{N}_j^{0,i}} T(\hat{\mathbf{z}}) - \sum_{\mathbf{z}\in\mathcal{N}_j^{-,i}} T(\mathbf{z})}{2^{D-1}\|\mathbf{v}_j^i\|} \quad (9)$$

which may also be expressed as:

$$\widehat{D}^{-\mathbf{v}_j^i}T(\mathbf{x}) = \frac{T(\mathbf{x}) - T_{\nu_j}^i}{\Delta_{\nu_j}^i} \quad (10)$$

$$T_{\nu_j}^i = \sum_{\mathbf{z}\in\mathcal{N}_j^{-,i}} T(\mathbf{z}) - \sum_{\substack{\hat{\mathbf{z}}\in\mathcal{N}_j^{0,i} \\ \hat{\mathbf{z}}\neq\mathbf{x}}} T(\hat{\mathbf{z}}) \quad (11)$$

$$\Delta_{\nu_j}^i = 2^{D-1}\|\mathbf{v}_j^i\| \quad (12)$$

This scheme provides second order accuracy for the derivatives. The proof for this is given in the Appendix. Equations (9) to (12) may also be applied to $D \neq 3$. $\mathcal{N}_j^{0,i}$ and $\mathcal{N}_j^{-,i}$ will have only one node for $D = 1$; and for $D = 2$, they will be the nodes belonging to respective parallel edges of a parallelogram. Equivalently, for $D > 3$, they will be the nodes of $D - 1$ dimensional parallelotopes.

Up to this point, we have assumed that all the required nodes for the scheme are already *Frozen*; however, this is not true in general. Therefore, an algorithm is needed to deal with the situations where one or more of the required nodes are not *Frozen*. Our strategy will be to try first the centered second order scheme, and if it cannot be applied, to revert to first order. The following algorithm is employed. Firstly, check how many stencil vectors lead to a *Frozen* node. If there is only one *Frozen* node, no more checks are required and the value can be computed with the proposed scheme. If there are two or more *Frozen* nodes, check if the rest of the "face" nodes required for the scheme are *Frozen*. If they are, then the proposed scheme is employed; and if not, the first order scheme is applied, using the available *Frozen* nodes.

B. Development for Non Orthogonal Stencils

The following exposition differs from [13], firstly, in that it does not assume $G^i = Id$ at any point, so it may be applied in any regular isotropic or anisotropic grids regardless of the orthogonality of the stencils. Like the original FM method, Eq. (2) is numerically computed by an upwind finite difference scheme. However, the cross terms must now be taken into account. The second difference from [13] is that the implementation of the upwind scheme is slightly modified. Specifically, notice that the forward approximation to $\langle\nabla T(\mathbf{x}), \mathbf{v}_j^i\rangle$ may also be seen as the backward approximation to $\langle\nabla T(\mathbf{x}), -\mathbf{v}_j^i\rangle$, and that $\langle\nabla T(\mathbf{x}), -\mathbf{v}_j^i\rangle = -\langle\nabla T(\mathbf{x}), \mathbf{v}_j^i\rangle$, in agreement with Eq. (4). Then, Eq. (4) may be reformulated as:

$$\psi_j^i(\mathbf{x}) \approx H(T(\mathbf{x}) - T_{\nu_j}^i) \cdot \sigma_j^i \cdot \left(\frac{T(\mathbf{x}) - T_{\nu_j}^i}{\Delta_{\nu_j}^i}\right) \quad (13)$$

where $H(z)$ is the Heaviside step function[2]; $T_{\nu_j}^i$ and $\Delta_{\nu_j}^i$ are

[2]$H(z) = 1$ for $z \geq 0$ and $H(z) = 0$ otherwise. Here, the Heaviside step function acts as a switch to prevent adding unsuitable contributions to Eq. (2). Note that $H^2(z) = H(z)$.

the equivalent values in Eqs. (11) and (12) respectively; and $\sigma_j^i = 1$ if the *Frozen* node is at $\mathbf{x} - \mathbf{v}_j^i$, otherwise $\sigma_j^i = -1$.

Thus, since the relationship between ∇T and the directional derivative is sensitive to the orientation, $\Psi^i(\mathbf{x}) = diag(\sigma^i)R^i\nabla T(\mathbf{x})$, with $diag(\sigma^i)$ a diagonal matrix with $\sigma^i = (\sigma_1^i, \ldots, \sigma_D^i)^T$ as its main diagonal. Taking into account that in this particular case $1/\sigma_j^i = \sigma_j^i$, and thus, $diag(\sigma^i)^{-1} = diag(\sigma^i)$, Eq. (2) becomes:

$$\Psi^i(\mathbf{x})^T \widehat{G}^i \Psi^i(\mathbf{x}) = \frac{1}{F(\mathbf{x})^2} = C(\mathbf{x})^2, \quad \text{s.t. } T(\Gamma_0) = 0 \quad (14)$$

where $\widehat{G}^i = diag(\sigma^i)^{-1} G^i diag(\sigma^i) = diag(\sigma^i) G^i diag(\sigma^i)$.

Plugging Eq. (13) into Eq. (14) and incorporating the derivative approximations proposed in Section III-A yields a quadratic equation for $T(\mathbf{x})$ of the form $\sum_{l=0}^{2} \widehat{\lambda}_l^i T(\mathbf{x})^l = 1/F(\mathbf{x}_c^i)^2$, the coefficients of which have the expression:

$$\widehat{\lambda}_l^i = \sum_{j=1}^{D} \frac{g_{jj}^i H_j^i}{(\Delta_{\nu_j}^i)^2} \alpha_{jl} + 2 \sum_{j=1}^{D-1} \sum_{k=j+1}^{D} \frac{g_{jk}^i \sigma_j^i \sigma_k^i H_j^i H_k^i}{\Delta_{\nu_j}^i \Delta_{\nu_k}^i} \beta_{jkl} \quad (15)$$

with $H_j^i = H(T(\mathbf{x}) - T_{\nu_j}^i)$, and:

$$\alpha_{j2} = 1 \qquad \beta_{jk2} = 1 \qquad (16)$$
$$\alpha_{j1} = -2T_{\nu_j}^i \qquad \beta_{jk1} = -(T_{\nu_j}^i + T_{\nu_k}^i) \qquad (17)$$
$$\alpha_{j0} = (T_{\nu_j}^i)^2 \qquad \beta_{jk0} = T_{\nu_j}^i T_{\nu_k}^i \qquad (18)$$

Since H_j^i depends on $T(\mathbf{x})$, the strategy followed is to firstly assign $H_j^i = 1$ if there is a known upwind node along $\pm\mathbf{v}_j^i$, and $H_j^i = 0$ otherwise. Then $T(\mathbf{x})$ is computed, and the H_j^i are consequently recalculated. If any of them modifies its value, the procedure is repeated until all H_j^i remain unchanged.

C. Stencil Set Design

For any stencil $\mathcal{S} = \{\mathbf{v}_i\}_{i=1}^{D}$, let us associate a matrix $S \in \mathbb{Z}^{D \times D}$, $S_{ij} = \tilde{v}_{i,j}$ to it, where $\tilde{\mathbf{v}}_i$ is the expression of $\mathbf{v}_i$ on an isotropic grid with $h = 1$ (that is, $\tilde{v}_{i,j} \in \mathbb{Z}$). There are stencils that yield the same value for $T(\mathbf{x})$ using Eq. (2), and thus we will refer to them as equivalent within the MSFM framework, or MSFM-equivalent.

Let $\mathcal{Q}_a = \{\mathcal{S}_a^i\}_{i=1}^{Q}$ be a stencil set. Then, we say that $\mathcal{Q}_a$ is invariant to permutation with respect to MSFM if for any permutation matrix P, $S_a^i P$ is MSFM-equivalent to one of the stencils in $\mathcal{Q}_a$. If we consider $\mathcal{Q}_o$, the 6-stencil set used in [13] for 3-D given in Table I, we observe that stencils $\mathcal{S}^{o5}$ and $\mathcal{S}^{o6}$ do not meet this condition, and therefore, the stencil set is not invariant to permutation. This is the cause of the error asymmetry observed in Fig. 2d. Thus, in order to be made permutation-invariant, $\mathcal{Q}_o$ should be extended by adding to the set the permutations of $\mathcal{S}^{o5}$ and $\mathcal{S}^{o6}$.

IV. NUMERICAL EXPERIMENTS

A. Methodology

To evaluate the accuracy of the method, several 2D and 3D analytical functions, listed on Table II, have been used. Several of the local costs on it are members of the parametric family of sinusoidal functions $f(z, \kappa, \tau) = \kappa - \cos(z/\tau)$, with their associated $u(z, \kappa, \tau) = \int_0^z f(t, \kappa, \tau)dt = \kappa z - \tau\sin(z/\tau)$.

Table II
2D AND 3D ANALYTICAL FUNCTIONS EMPLOYED IN THE EXPERIMENTS. SEE MAIN TEXT FOR THE DEFINITION OF FUNCTION $u(z, \kappa, \tau)$.

i	$\mathbf{T_i(x)}$	$\mathbf{F_i(x) = (C_i(x))^{-1}}$
1	$\sqrt{x^2 + y^2}$	1
2	$\frac{x^2}{25} + \frac{y^2}{9}$	$\left(2\sqrt{\frac{x^2}{625} + \frac{y^2}{81}}\right)^{-1}$
3	$\frac{x^2}{100} + \frac{y^2}{20}$	$\left(2\sqrt{\frac{x^2}{10^4} + \frac{y^2}{400}}\right)^{-1}$
4	$u(T_1(x,y), 1, 2)$	$(f(T_1(x,y), 1, 2))^{-1}$
5	$u(T_1(x,y), 1, 8)$	$(f(T_1(x,y), 1, 8))^{-1}$
6	$\sqrt{x^2 + y^2 + z^2}$	1
7	$\frac{x^2}{25} + \frac{y^2}{16} + \frac{z^2}{36}$	$\left(2\sqrt{\frac{x^2}{625} + \frac{y^2}{256} + \frac{z^2}{1296}}\right)^{-1}$
8	$\frac{x^2}{100} + \frac{y^2}{20} + \frac{z^2}{20}$	$\left(2\sqrt{\frac{x^2}{10^4} + \frac{y^2}{400} + \frac{z^2}{400}}\right)^{-1}$
9	$u(T_6(x,y,z), 9/8, 2)$	$(f(T_6(x,y,z), 9/8, 2))^{-1}$
10	$u(T_6(x,y,z), 9/8, 20)$	$(f(T_6(x,y,z), 9/8, 20))^{-1}$

These functions will help us inspect the behavior of the schemes with respect to the local cost maximum frequency. The error between two realizations $T_{r1}(\mathbf{x})$ and $T_{r2}(\mathbf{x})$ is quantified by their absolute deviation norms L_1, L_∞ and L_∞^n, where n stands for normalized, and they are defined as:

$$L_1(T_{r1}(\mathbf{x})), T_{r2}(\mathbf{x})) = \frac{\sum_{i=1}^{N} |T_{r1}(\mathbf{x}_i) - T_{r2}(\mathbf{x}_i)|}{N} \quad (19)$$

$$L_\infty(T_{r1}(\mathbf{x})), T_{r2}(\mathbf{x})) = \max_{i=1,\ldots,N} (|T_{r1}(\mathbf{x}_i) - T_{r2}(\mathbf{x}_i)|) \quad (20)$$

$$L_\infty^n(T_{r1}(\mathbf{x})), T_{r2}(\mathbf{x})) = \max_{i=1,\ldots,N} \frac{|T_{r1}(\mathbf{x}_i) - T_{r2}(\mathbf{x}_i)|}{\max(T_a(\mathbf{x}_i), \epsilon)} \quad (21)$$

where N is the number of nodes in the image, $\epsilon > 0$ is used to prevent divisions by zero and $T_a(\mathbf{x})$ the analytical solution. If no $T_{r2}(\mathbf{x})$ is specified, $T_{r2}(\mathbf{x}) = T_a(\mathbf{x})$ is assumed; then, the previous norms measure the absolute error yielded by the method employed for generating $T_{r1}(\mathbf{x})$.

Several stencil sets have been used in the experiments. For 2-D, the stencil sets employed in the experiments consist of a number of the orthogonal stencils given in Table III. These stencil subsets are selected so that the maximum square radius ρ^2 of the vectors composing the stencil is below a certain threshold m. The same has been done in 3-D with the orthogonal stencils given in Table IV. These stencil sets are referred to as $\mathcal{Q}_m$. Note that $\mathcal{Q}_m \subset \mathcal{Q}_n$, if $1 \leq m \leq n$. Additionally, the 6-stencil set $\mathcal{Q}_o$ proposed in the original MSFM method (see Table I) is included in several experiments, along with two more sets including non-orthogonal stencils. The first one, $\mathcal{Q}_e$, is an extension of the 6-stencil set, where permutations of $\mathcal{S}^{o5}$ and $\mathcal{S}^{o6}$ have been included to make it invariant with respect to permutations. The second one, $\mathcal{Q}_d$, is designed so that the vectors within a stencil, as Fig. 5 shows, point from a node to three of its 26-neighbors, delimiting a triangle embedded in one of the faces of the cube defined by the 26-neighborhood. Specifically, each stencil is composed, as indicated in Fig. 5, of three vectors that point respectively to a face center, an edge center and a vertex of the cube. Thus, each cube face contains eight such stencils. Notice that for this stencil set,
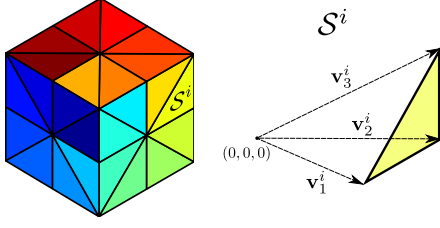
Figure 5. Layout of the $\mathcal{Q}_d$ stencils. Each stencil is represented by a distinctly colored triangle, the vertices of which have as coordinates the result of applying the respective stencil vectors to the cube center $(0,0,0)$.

Table III
ORTHOGONAL 2-D STENCILS IN AN ISOTROPIC GRID WITH $h = 1$. ρ^2 IS THE MAXIMUM SQUARE NORM OF THE STENCIL VECTORS.

i	ρ^2	$\mathcal{S}^i$
1	1	$(1,0)^T, (0,1)^T$
2	2	$(1,1)^T, (1,-1)^T$
3	5	$(2,1)^T, (1,-2)^T$
4	5	$(2,-1)^T, (1,2)^T$
5	10	$(3,1)^T, (1,-3)^T$
6	10	$(3,-1)^T, (1,3)^T$

Table IV
ORTHOGONAL 3-D STENCILS IN AN ISOTROPIC GRID WITH $h = 1$. ρ^2 IS THE MAXIMUM SQUARE NORM OF THE STENCIL VECTORS.

i	ρ^2	$\mathcal{S}^i$
1	1	$(0,0,1)^T, (1,0,0)^T, (0,1,0)^T$
2	2	$(0,0,1)^T, (1,1,0)^T, (1,-1,0)^T$
3	2	$(1,0,0)^T, (0,1,1)^T, (0,-1,1)^T$
4	2	$(0,1,0)^T, (1,0,1)^T, (-1,0,1)^T$
5	5	$(0,0,1)^T, (2,1,0)^T, (1,-2,0)^T$
6	5	$(0,0,1)^T, (2,-1,0)^T, (1,2,0)^T$
7	5	$(1,0,0)^T, (0,1,2)^T, (0,-2,1)^T$
8	5	$(1,0,0)^T, (0,-1,2)^T, (0,2,1)^T$
9	5	$(0,1,0)^T, (1,0,2)^T, (-2,0,1)^T$
10	5	$(0,1,0)^T, (-1,0,2)^T, (2,0,1)^T$
11	6	$(1,0,1)^T, (-1,1,1)^T, (-1,-2,1)^T$
12	6	$(1,0,1)^T, (-1,-1,1)^T, (-1,2,1)^T$
13	6	$(0,1,1)^T, (1,-1,1)^T, (-2,-1,1)^T$
14	6	$(0,1,1)^T, (-1,-1,1)^T, (2,-1,1)^T$
15	6	$(0,-1,1)^T, (1,1,1)^T, (-2,1,1)^T$
16	6	$(0,-1,1)^T, (-1,1,1)^T, (2,1,1)^T$
17	6	$(-1,0,1)^T, (1,1,1)^T, (1,-2,1)^T$
18	6	$(-1,0,1)^T, (1,-1,1)^T, (1,2,1)^T$
19	6	$(1,1,0)^T, (1,-1,1)^T, (-1,1,2)^T$
20	6	$(1,1,0)^T, (-1,1,1)^T, (1,-1,2)^T$
21	6	$(1,-1,0)^T, (1,1,1)^T, (-1,-1,2)^T$
22	6	$(1,-1,0)^T, (-1,-1,1)^T, (1,1,2)^T$
23	9	$(2,1,2)^T, (-1,-2,2)^T, (-2,2,1)^T$
24	9	$(2,-1,2)^T, (-1,2,2)^T, (-2,-2,1)^T$
25	9	$(1,2,2)^T, (-2,-1,2)^T, (2,-2,1)^T$
26	9	$(1,-2,2)^T, (-2,1,2)^T, (2,2,1)^T$

each of the stencils contained in one cube face is MSFM-equivalent to one of the stencils contained in the opposite cube face. Therefore, only the stencils belonging to three orthogonal faces of the cube are incorporated to $\mathcal{Q}_d$. This set is invariant to permutations and is composed of 24 highly non-orthogonal stencils (see Fig. 5).

In Section IV-B, the accuracy of the proposed MSFM scheme is evaluated and compared with that of the original MSFM method for the 2D and 3D analytical functions. In Section IV-C, the convergence of the MSFM methods is evaluated with the 2D analytical functions. The experiment is not repeated for the 3D functions, for the sake of conciseness and due to the guarantee given by the results in the Appendix. In Section IV-D, the accuracy and the invariance against axes permutations are assessed with the 3D functions. Since the stencils employed in 2D are all orthogonal, the invariance to permutations does not need to be tested.

B. Accuracy

1) Experiments with 2D Functions: The accuracy of the MSFM using the original first order, second order, and the proposed schemes (respectively referred to as MS_1, MS_2 and MS_c) for the 2-D analytical functions was assessed using an isotropic grid of 101×101 pixels with spacing $h = 1$ and $-50 \leq x, y \leq 50$, with the front propagation starting from $\Gamma_0 = (0,0)^T$. The experiment was performed using different sets composed by the stencils given in Table III with a radius ρ below a given value.

The L_1 and L_∞ error norms applying MS_1, MS_2 and MS_c to the analytical 2-D functions $T_1(\mathbf{x})$ to $T_5(\mathbf{x})$ for $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$ are given in Table V. Since MS_1 and MS_2 are originally only proposed for $\mathcal{Q}_1$ and $\mathcal{Q}_2$, these results are italicized. In every experiment, the proposed scheme yielded equal or more accurate results than both the original first and second order schemes. For $T_1(\mathbf{x})$, where the local cost is constant, MS_c provides results with significantly lower error than the original schemes, especially for $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$. It is worth mentioning that with the latter stencil sets, MS_2 behaves worse than MS_1. The reason for this is that there were cases for $\mathcal{S}^3$ and $\mathcal{S}^4$ when the second *Frozen* node was not in the domain of dependence of the considered node. Nevertheless, note that $\mathcal{S}^3$ and $\mathcal{S}^4$ were not proposed in the original method described in [13]; we are simply checking its performance when those stencils are used. The errors for the elliptical functions $T_2(\mathbf{x})$ and $T_3(\mathbf{x})$ show that MS_1 yields the same error for all the stencil sets. The reason for this is that the stencil $\mathcal{S}^1$ is chosen at almost every node, because vectors of higher norm overestimate the cost. In contrast, the second order schemes are more accurate and are able to reduce the error by increasing the stencil set, with the proposed scheme yielding more accurate results. Both $T_4(\mathbf{x})$ and $T_5(\mathbf{x})$, which have the same functional expression, have radially periodic local costs with period 4π and 16π for $C_4(\mathbf{x})$ and $C_5(\mathbf{x})$, respectively. The accuracy of the MSFM schemes decreases as more stencils are considered, except MS_1 for $T_5(\mathbf{x})$ with $\mathcal{Q}_2$. The proposed scheme achieves the minimum error with all stencil sets for $T_4(\mathbf{x})$ and $T_5(\mathbf{x})$.

2) Experiments with 3D Functions: The same accuracy experiment was carried out for the 3-D analytical functions $T_6(\mathbf{x})$ to $T_{10}(\mathbf{x})$, on a $51 \times 51 \times 51$ isotropic grid with $h = 1$ and bounds $-25 \leq x, y, z \leq 25$. The propagation starts from $\Gamma_0 = (0,0,0)^T$. The errors incurred by the two original and the proposed MSFM methods with different stencil sets are given in Table VI.

Table V
L_1 AND L_∞ NORMS FOR $T_1(\mathbf{x})$ TO $T_5(\mathbf{x})$ USING THE ORIGINAL AND THE PROPOSED MSFM SCHEMES WITH STENCIL SETS OF VARYING SQUARE RADIUS ρ.

Norm/$\mathcal{Q}_\rho$	T_1			T_2			T_3			T_4			T_5		
	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$
$L_1/\mathcal{Q}_1$	0.746	0.251	0.044	3.815	0.359	0.082	1.515	0.122	0.032	0.661	1.318	0.079	0.643	0.259	0.020
$L_1/\mathcal{Q}_2$	0.292	0.228	0.042	3.815	0.090	0.013	1.515	0.022	0.005	0.931	3.301	0.335	0.539	0.549	0.035
$L_1/\mathcal{Q}_5$	0.131	1.200	0.005	3.815	0.139	0.010	1.515	0.024	0.003	3.210	5.676	0.841	0.727	0.721	0.055
$L_1/\mathcal{Q}_{10}$	0.092	1.702	0.005	3.815	0.148	0.009	1.515	0.030	0.004	6.018	9.699	2.460	1.165	1.372	0.165
$L_\infty/\mathcal{Q}_1$	1.315	0.370	0.077	7.556	0.644	0.114	3.000	0.174	0.045	1.440	2.521	0.131	1.379	0.712	0.031
$L_\infty/\mathcal{Q}_2$	0.578	0.307	0.077	7.556	0.253	0.065	3.000	0.097	0.026	2.700	6.734	0.879	1.280	1.316	0.063
$L_\infty/\mathcal{Q}_5$	0.287	1.307	0.016	7.556	0.245	0.063	3.000	0.091	0.025	6.669	10.771	1.523	1.272	1.549	0.120
$L_\infty/\mathcal{Q}_{10}$	0.216	1.845	0.016	7.556	0.245	0.049	3.000	0.091	0.023	11.324	17.606	4.447	1.986	2.649	0.339

For the unitary local cost $C_6(\mathbf{x})$, the proposed scheme achieves the best results both for L_1 and L_∞, except for $\mathcal{Q}_d$, where the first order scheme yields the lowest errors for all the considered $T_i(\mathbf{x})$. For $T_6(\mathbf{x})$, MS$_2$ has lower errors than MS$_1$ for $\mathcal{Q}_1$ and $\mathcal{Q}_2$, but unlike MS$_c$, has lower performance in the rest of the stencil sets. For the ellipsoidal functions $T_7(\mathbf{x})$ and $T_8(\mathbf{x})$, the second order schemes perform better than the first order scheme for all stencil sets except $\mathcal{Q}_d$. We observe that in this case, MS$_2$ yields a slightly lower error than MS$_c$ for $\mathcal{Q}_1$ and $\mathcal{Q}_2$, and also in $\mathcal{Q}_5$ for $T_7(\mathbf{x})$; for the rest of the stencil sets, excepting $\mathcal{Q}_d$, MS$_c$ achieved the best results. For the fastest varying sinusoidal function $T_9(\mathbf{x})$, the most accurate estimation among all method and stencil set combinations was achieved by MS$_c$ with $\mathcal{Q}_1$. For $\mathcal{Q}_2$, MS$_c$ also yielded the lowest error. Excluding $\mathcal{Q}_1$ and $\mathcal{Q}_2$, the MS$_1$ method is the most accurate one with the rest of the stencil sets, followed by MS$_c$ in all cases except for $\mathcal{Q}_d$, where MS$_c$ incurred in the highest error. For the slowest varying sinusoidal function $T_{10}(\mathbf{x})$, the second order schemes MS$_2$ and MS$_c$ provided better accuracy than MS$_1$ for all the stencil sets except, again, for $\mathcal{Q}_d$. The lowest error for $T_{10}(\mathbf{x})$ was achieved by combining MS$_c$ with $\mathcal{Q}_5$. MS$_c$ performed better than MS$_2$ for $\mathcal{Q}_5$, $\mathcal{Q}_6$, $\mathcal{Q}_9$, $\mathcal{Q}_o$ and $\mathcal{Q}_e$; for $\mathcal{Q}_1$ and $\mathcal{Q}_2$, both second order methods performed similarly. Remark as well that MS$_c$ achieved lower errors than MS$_2$ for the non-orthogonal stencil sets $\mathcal{Q}_o$ and $\mathcal{Q}_e$. In order to a better assessment of the influence of τ over the accuracy of the different methods, the experiment was repeated varying τ between 0.4 and 90. Figure 6 shows the L_1 norm against τ for all methods and stencil sets. Notice that τ is inversely proportional to the sinusoidal frequency. We observe that, again except $\mathcal{Q}_d$, MS$_c$ generally achieves the lowest errors for most of the considered τ values, both low and high. We can see as well that for low τ (high frequency), the best results are achieved with one stencil, while for high τ (low frequency), larger stencil sets yield lower errors.

C. Convergence

In order to experimentally test the rate of convergence of the numerical scheme, a set of domains was created by discretizing $[-1,1] \times [-1,1]$ with successive diminishing grid spacings $h^k = 2^{-k}, k = 5, \ldots, 9$. Then, MS$_c$ was applied to $T_1(\mathbf{x})$, $T_2(\mathbf{x})$ and $T_4(\mathbf{x})$ with $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$, starting from the node $(0,0)^T$. For each experiment, the L_1 norm was computed. The results are shown in Fig. 7. The line colors for MS$_1$, MS$_2$ and MS$_c$ are blue, green and red, respectively; in addition, the line marker shape is chosen according to the stencil set employed: a solid line for $\mathcal{Q}_1$, and a plus sign, a circle and an asterisk for $\mathcal{Q}_2$, $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$, respectively. It may be observed that in all cases the error decreases as h gets smaller. In addition, the lowest error values are achieved by the proposed method. It may also be observed that the second order schemes have a steeper slope than MS$_1$, which means a faster convergence to the exact solution, and that the slopes of the second order methods are very similar. For $T_2(\mathbf{x})$ and $T_4(\mathbf{x})$, MS$_1$ yields almost identical results with the four stencil sets. The reason for this is that $\mathcal{S}^1$ is chosen at almost every node. It may also be observed that for $T_1(\mathbf{x})$ and MS$_2$ with the stencil sets $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$ are less accurate than MS$_2$ with the stencil sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$; while for MS$_1$ and MS$_c$, using more stencils decreases the error of the solution.

D. Invariance with Respect to Axes Permutations

The influence of permuting the stencil vectors in $\mathcal{Q}_o$ was assessed over a rotated ellipsoidal function:

$$T_{\mathrm{r}}(x,y,z,\theta,\phi) = [x,y,z]R(\theta,\phi)\Lambda R(\theta,\phi)^T[x,y,z]^T \quad (22)$$

where Λ is a diagonal matrix with diagonal elements $(\frac{1}{36}, \frac{1}{4}, \frac{1}{4})$ and $R(\theta,\phi) = R_{\mathbf{u}}(\phi)R_{\mathbf{z}}(\theta)$ composes a counterclockwise rotation of θ around the z-axis and then applies a counterclockwise rotation of ϕ around $\mathbf{u} = R_{\mathbf{z}}(\theta)\mathbf{y}$. Again, a $51 \times 51 \times 51$ isotropic grid with $h = 1$ and $-25 \leq x,y,z \leq 25$ was employed.

For several different rotation angles θ and ϕ, $T_{\mathrm{r}}(x,y,z,\theta,\phi)$ was estimated using MS$_1$, MS$_2$ and MS$_c$ with the original $\mathcal{Q}_o$ stencil set as well as with the remaining 5 possible permutations of each vector component in every stencil of each stencil set (the same permutation matrix was applied to every vector). Then, the L_∞^{n} norm was applied to the absolute deviations between pairs of different permutations, which gives rise to $\binom{6}{2} = 15$ possible combinations. In Table VII, the worst case absolute deviations between pairs (among all possible permutations) using MS$_c$ with $\mathcal{Q}_o$ are shown as a percentage of the analytical value $T_{\mathrm{r}}(x,y,z,\theta,\phi)$, and their values range between 21.09% and 56.96%. This shows that the variation in the results by simply permuting

Table VI

L_1 AND L_∞ NORMS FOR $T_6(\mathbf{x})$ TO $T_{10}(\mathbf{x})$ USING THE ORIGINAL AND THE PROPOSED MSFM SCHEMES WITH STENCIL SETS OF VARYING RADIUS ρ.

Norm/$\mathcal{Q}_\rho$	T_6			T_7			T_8			T_9			T_{10}		
	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$	MS$_1$	MS$_2$	MS$_c$
$L_1/\mathcal{Q}_1$	1.18	0.61	0.26	1.66	0.33	0.38	1.40	0.26	0.29	1.53	0.39	0.13	0.56	0.12	0.12
$L_1/\mathcal{Q}_2$	0.63	0.39	0.08	1.66	0.04	0.07	1.40	0.05	0.06	0.57	0.70	0.40	0.52	0.02	0.02
$L_1/\mathcal{Q}_5$	0.51	1.30	0.04	1.66	0.03	0.04	1.40	0.04	0.03	1.04	2.55	1.63	0.52	0.10	0.01
$L_1/\mathcal{Q}_6$	0.25	1.24	0.05	1.66	0.02	0.02	1.40	0.03	0.01	1.47	3.42	2.11	0.50	0.10	0.01
$L_1/\mathcal{Q}_9$	0.20	1.64	0.05	1.66	0.03	0.02	1.40	0.03	0.01	2.38	4.62	2.91	0.50	0.12	0.02
$L_1/\mathcal{Q}_o$	0.52	1.29	0.31	1.12	1.02	0.36	0.98	2.16	0.06	0.80	2.49	1.12	0.38	0.35	0.13
$L_1/\mathcal{Q}_e$	1.19	2.45	0.64	0.82	2.61	0.59	1.00	2.42	0.44	1.32	3.64	1.59	0.23	0.88	0.23
$L_1/\mathcal{Q}_d$	7.28	7.71	10.49	8.25	9.30	11.15	6.87	7.76	9.32	8.36	9.28	12.18	2.82	3.17	3.87
$L_\infty/\mathcal{Q}_1$	1.90	0.76	0.62	3.26	0.56	1.52	2.75	0.52	1.20	2.63	0.91	0.59	1.27	0.19	0.52
$L_\infty/\mathcal{Q}_2$	1.06	0.48	0.29	3.26	0.20	0.51	2.75	0.17	0.36	1.78	1.59	1.12	1.21	0.04	0.18
$L_\infty/\mathcal{Q}_5$	1.00	1.69	0.23	3.26	0.20	0.09	2.75	0.16	0.07	2.65	4.94	3.69	1.21	0.14	0.05
$L_\infty/\mathcal{Q}_6$	0.46	1.34	0.17	3.26	0.16	0.05	2.75	0.14	0.05	3.72	6.33	4.37	1.18	0.11	0.05
$L_\infty/\mathcal{Q}_9$	0.37	1.98	0.15	3.26	0.16	0.05	2.75	0.14	0.05	6.01	8.50	6.94	1.18	0.14	0.04
$L_\infty/\mathcal{Q}_o$	3.25	5.07	2.48	3.34	6.36	3.16	5.90	8.22	0.54	3.75	6.45	2.88	1.49	2.68	1.82
$L_\infty/\mathcal{Q}_e$	3.26	5.07	2.48	4.93	7.96	3.17	5.90	8.22	2.30	3.79	6.78	3.21	1.49	2.67	1.82
$L_\infty/\mathcal{Q}_d$	12.26	12.75	18.58	24.66	26.68	34.69	21.07	22.77	32.57	13.88	15.72	21.41	8.60	9.40	12.79
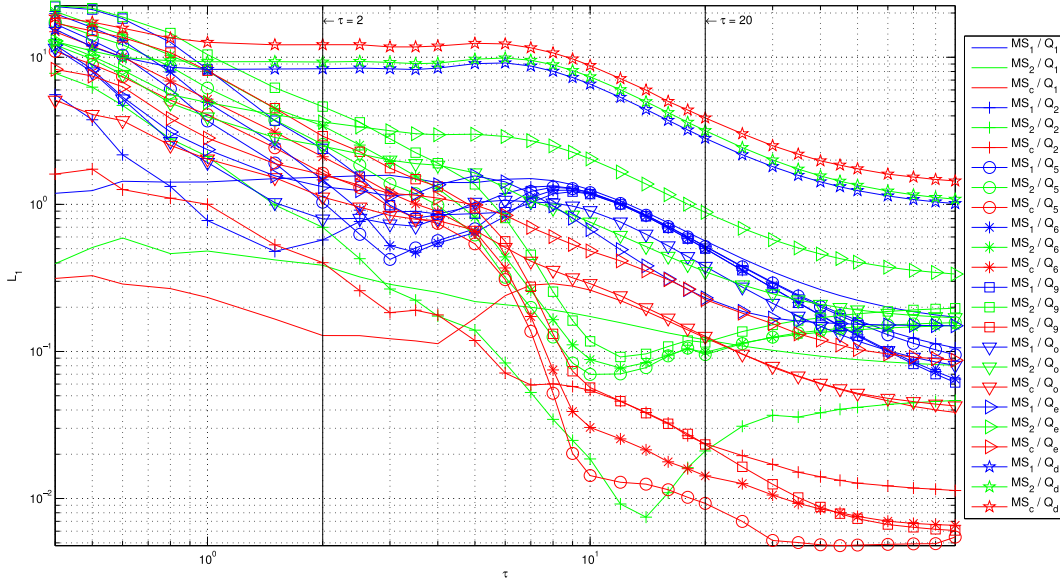
Figure 6. L_1 norm yielded by MS$_1$, MS$_2$, and MS$_c$ with a sinusoidal cost of varying τ, for each of the considered stencil sets.
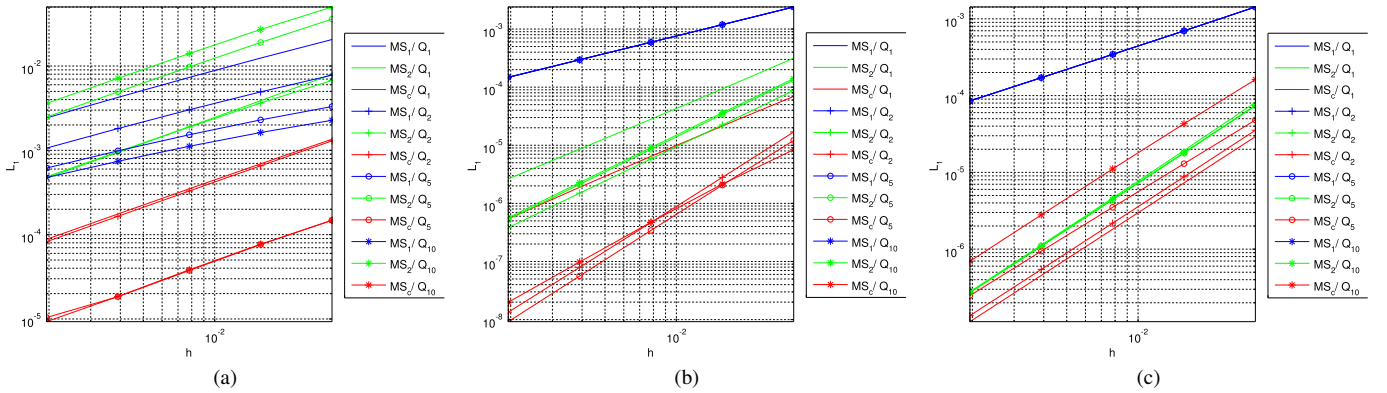
Figure 7. L_1 norm with respect to the grid spacing h for (a) $T_1(\mathbf{x})$, (b) $T_2(\mathbf{x})$ and (c) $T_4(\mathbf{x})$, using MS$_1$ (blue), MS$_2$ (green) and MS$_c$ (red), with the stencil sets $\mathcal{Q}_1$ (solid line), $\mathcal{Q}_2$ (plus markers), $\mathcal{Q}_5$ (circle markers) and $\mathcal{Q}_{10}$ (asterisk markers).

the stencil vector coordinates in a non permutation-invariant stencil set may be quite significant. The proposed scheme is more sensitive to non permutation-invariant stencils due to the higher participation of the stencil vectors in the estimation of the directional derivatives. However, when the experiment was repeated using the permutation-invariant stencil sets, the worst

Table VII
$100 L_\infty^n$ NORM OF THE LOCAL DIFFERENCE BETWEEN COMPUTATIONS OF $T_r(x, y, z, \theta, \phi)$ WITH DIFFERENT PERMUTATIONS OF $\mathcal{Q}_o$.

	$\phi = 21°$	$\phi = 39°$	$\phi = 57°$	$\phi = 75°$
$\theta = 21°$	42.55	35.10	41.47	29.71
$\theta = 39°$	48.47	43.48	43.92	26.19
$\theta = 57°$	27.49	56.96	28.78	37.49
$\theta = 75°$	28.54	33.85	44.75	21.09

case absolute deviations were below 10^{-12} in all cases.

The previous experiment was performed again with the 3D analytical functions, $T_6(\mathbf{x})$ to $T_{10}(\mathbf{x})$, to test the invariance with respect to axes permutation of the stencil sets with MS_c. These functions were estimated with $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_5$, $\mathcal{Q}_6$, $\mathcal{Q}_9$, $\mathcal{Q}_o$, $\mathcal{Q}_e$, and $\mathcal{Q}_d$. For each function and stencil set, the experiment was again carried out applying the proposed MSFM method with each of the six possible axis permutations of the stencil set. Then, the L_1 norms of the absolute difference between the 15 pairs of permutations were computed and then averaged.

However, the results may be influenced by the underlying format employed for the data value representation. It is well known that the addition and product operation of values with floating point format (FL) is not associative [18]. This introduces small differences when computing $T(\mathbf{x})$ with equivalent stencils, and the effect on the algorithm of those differences may be compounded by the min and max operations involved. On the other hand, the fixed point format (FI) fulfills the associative property for the addition and the product, and thus it does not suffer from the previous pitfalls; but compared to the floating point format, its range and accuracy is much more restricted.

Table VIII provides the results for both aforementioned formats. The computation of the different $T_i(\mathbf{x})$ is carried out using each of the formats and then converted to floating point. The fixed point arithmetic was implemented using the MATLAB 2014a Fixed-Point Designer [19]. We can see that with the FL format, the permutation invariant stencil sets achieve very low errors ($< 10^{-13}$) in 22 out of 35 cases, with the remaining values between $1.6 \cdot 10^{-6}$ and $1.6 \cdot 10^{-3}$. The highest differences between permutations are yielded by $\mathcal{Q}_o$, with errors ranging from $2.6 \cdot 10^{-2}$ to 0.13, and at least 75 times higher than the maximum error committed by a permutation invariant stencil for the same analytical function. The FI format yields identical solutions to $T_6(\mathbf{x})$–$T_{10}(\mathbf{x})$ with all the permutation-invariant stencil sets. The non permutation invariant $\mathcal{Q}_o$ yields similar errors for FI and FL, which leads us to believe that the errors committed with FL and the permutation invariant stencil sets are induced by the data value representation.

The same experiment was carried out for $\mathcal{Q}_2$ (as representation of the orthogonal stencil sets), $\mathcal{Q}_o$, $\mathcal{Q}_e$ and $\mathcal{Q}_d$, by applying MS_c to $T_6(\mathbf{x})$–$T_{10}(\mathbf{x})$ with the 2^3 possibilities of sign changes in the stencil vectors for every considered stencil set, instead of an axes permutation. We observe that $\mathcal{Q}_2$, $\mathcal{Q}_o$ and $\mathcal{Q}_e$ provided identical results with all possible vector sign changes for the considered functions, both with FL and FI.

However, $\mathcal{Q}_d$ shows errors $> 10^{-15}$ for $T_6(\mathbf{x})$, $T_9(\mathbf{x})$ and $T_{10}(\mathbf{x})$, even with the FI format. These differences seem to originate in nodes where there are two *Frozen* neighbors along the same stencil vector.

V. CASE OF USE: MINIMUM COST PATHS

In this section, we describe a use case where the MSFM method is used for optimal path planning in UAV flood monitoring missions. In this case study, path planning is carried out considering the limited battery life of the UAV and the need of close inspections of areas with flooded regions.

Initial flood maps obtained from MODIS hyperspectral data [20] are used to define cost functions where dry regions, watercourses and flooded areas are assigned different costs so that the preferable regions are the flooded ones. By using nonconstant costs, the problem solution should provide a trade-off between the actual travelled distance and the amount of flooded surface covered by the drone. Figure 8 shows the local cost maps $M_i(\mathbf{x})$, $i = 1, 2, 3$, defined for three evaluated test images. In our experiments, $M_i(\mathbf{x}) = 1$ in dry areas (white), $M_i(\mathbf{x}) = 0.75$ (gray) in rivers and known watercourses, and $M_i(\mathbf{x}) = 0.25$ (black) in flooded areas. Watercourses are assigned an intermediate cost, since they are interesting to inspect (to watch evolution), but not so important as actual flooded land.

The problem is posed as, given a starting and an ending point of the flight, find the path with minimum cost in between. To test performance, twenty points $\mathbf{y}_i, i = 1, \ldots, 20$ have been randomly selected on each image and pair-wise optimal paths have been determined. Figure 8 shows the selected random locations associated with each of the test images as colour filled circles.

Let $\Gamma_{ij}^{k, MS_\alpha}(l)$ denote the l-th entry in the minimum cost path on the local cost map $M_k(\mathbf{x})$, starting from $\mathbf{y}_i$ and ending at $\mathbf{y}_j$, that has been computed using the MSFM scheme $MS_\alpha = \{MS_1, MS_2, MS_c\}$ (See Fig. 9). $\Gamma_{ij}^{k, MS_\alpha}$ consists of N_{ij}^{k, MS_α} vertices, with $\Gamma_{ij}^{k, MS_\alpha}(1) = \mathbf{y}_i$ and $\Gamma_{ij}^{k, MS_\alpha}(N_{ij}^{k, MS_\alpha}) = \mathbf{y}_j$.

The procedure to compute $\Gamma_{ij}^{k, MS_\alpha}$ is similar to the methodology proposed in [10], [21]. Firstly, the MS_α scheme is used to compute $T(\mathbf{x})$ out from the local cost map $M_k(\mathbf{x})$, with $T(\mathbf{y}_j) = 0$. Then, the path from $\mathbf{y}_i$ to $\mathbf{y}_j$ is computed using a gradient descent algorithm on $T(\mathbf{x})$. The candidate path segments that subtend an angle larger than $7\pi/6$ with the preceding one are removed. This method may end early if there are local minima (originated from numerical issues) along the path. To compensate for this, if the path ends too far from $\mathbf{y}_j$, an additional step is made towards $\mathbf{y}_j$ and the gradient descent is restarted from that location. This is repeated a maximum number of times (200 in our experiments). If, after this, the path has still ended before reaching $\mathbf{y}_j$, a straight line segment from the path end to $\mathbf{y}_j$ is appended to $\Gamma_{ij}^{k, MS_\alpha}$.

The paths between all pairs $(\mathbf{y}_i, \mathbf{y}_j)$ of the randomly selected locations were computed for every $M_k(\mathbf{x})$, using the MS_1, MS_2 and MS_c schemes with $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_5$ and $\mathcal{Q}_{10}$. Remark that $T(\mathbf{x})$ only needs to be computed once for each $\mathbf{y}_j$, MSFM scheme and stencil set. Not all paths reached

Set	$T_6(\mathbf{x})$ FL	FI	$T_7(\mathbf{x})$ FL	FI	$T_8(\mathbf{x})$ FL	FI	$T_9(\mathbf{x})$ FL	FI	$T_{10}(\mathbf{x})$ FL	FI
$\mathcal{Q}_1$	2.4e-14	0	4.4e-15	0	1.9e-15	0	8.1e-14	0	1.9e-15	0
$\mathcal{Q}_2$	1.3e-14	0	3.0e-15	0	2.3e-15	0	9.2e-14	0	1.4e-15	0
$\mathcal{Q}_5$	1.1e-14	0	2.7e-15	0	1.7e-15	0	5.6e-04	0	4.4e-05	0
$\mathcal{Q}_6$	7.7e-15	0	2.0e-15	0	2.2e-05	0	2.8e-06	0	8.2e-06	0
$\mathcal{Q}_9$	1.6e-06	0	1.9e-15	0	1.9e-05	0	1.6e-03	0	2.2e-05	0
$\mathcal{Q}_o$	**1.3e-01**	**1.3e-01**	**6.1e-02**	**6.1e-02**	**5.7e-02**	**5.7e-02**	**1.2e-01**	**1.2e-01**	**2.6e-02**	**2.6e-02**
$\mathcal{Q}_e$	7.8e-05	0	1.5e-15	0	1.1e-15	0	9.5e-14	0	8.3e-16	0
$\mathcal{Q}_d$	6.0e-05	0	9.3e-16	0	8.9e-16	0	1.7e-06	0	1.2e-05	0

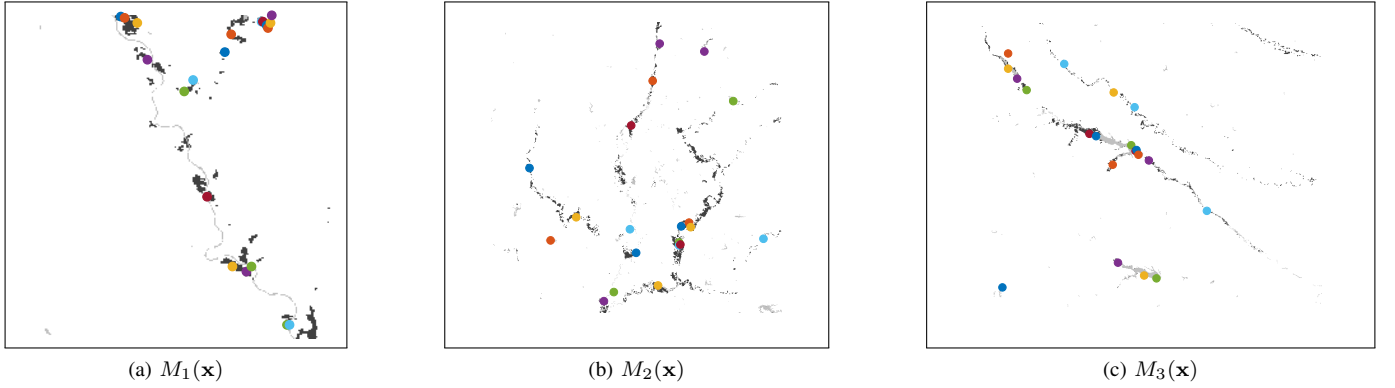(a) $M_1(\mathbf{x})$ (b) $M_2(\mathbf{x})$ (c) $M_3(\mathbf{x})$

Figure 8. MODIS flood map local costs used in the experiments, with their selected 20 random locations annotated with circles of distinct colors.
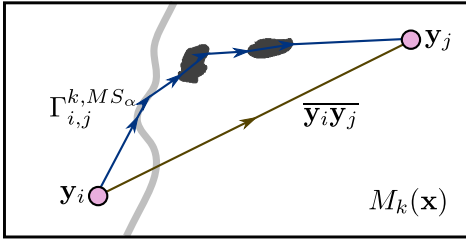

Figure 9. Illustration of the symbol notation for path trajectories.

the destination point after the gradient descent; a path was considered successful if the gradient descent ended at less than 3 pixels from $\mathbf{y}_j$. The mean distance in pixels to the destination point was computed for each method and stencil set, and the results are given in table IX. It may be observed that MS_c generates the paths that end closest to the destination point. For MS_1 and MS_2, only $\mathcal{Q}_1$ provides a mean distance under 3 pixels, whereas for MS_c, $\mathcal{Q}_1$ and $\mathcal{Q}_2$ yields mean distances of 0.21 and 2.01 pixels, respectively.

Table IX
MEAN DISTANCE IN PIXELS TO THE DESTINATION POINT AFTER THE GRADIENT DESCENT BY STENCIL SET AND MSFM SCHEME.

	$\mathcal{Q}_1$	$\mathcal{Q}_2$	$\mathcal{Q}_5$	$\mathcal{Q}_{10}$
MS_1	0.22	126.34	164.13	184.51
MS_2	0.22	79.65	143.95	177.92
MS_c	0.21	2.01	50.34	162.51

The goodness of the solutions is quantified as follows; our test set consists of all distinct origin-destination pairs $\{\mathbf{y}_i, \mathbf{y}_j, M_k(\mathbf{x})\}, i < j$, and the paths in between that result from the algorithms, provided that they have been successfully generated by MS_1, MS_2 and MS_c both forwards ($\Gamma_{ij}^{k,MS_\alpha}$) and backwards ($\Gamma_{ji}^{k,MS_\alpha}$). The following parameters were computed on them:

$$\gamma_{ijl}^{k,MS_\alpha} = \|\Gamma_{ij}^{k,MS_\alpha}(l) - \Gamma_{ij}^{k,MS_\alpha}(l-1)\| \quad (23)$$

$$MP_{ijl}^{k,MS_\alpha}(\zeta(\cdot)) = \frac{\zeta(\Gamma_{ij}^{k,MS_\alpha}(l)) + \zeta(\Gamma_{ij}^{k,MS_\alpha}(l-1))}{2} \quad (24)$$

$$D_{ij}^{k,MS_\alpha} = \sum_{l=2}^{N_{ij}^{k,MS_\alpha}} \gamma_{ijl}^{k,MS_\alpha} \quad (25)$$

where $\|\cdot\|$ is the Euclidean norm, $\zeta(\cdot)$ is some function defined over the image ($M_k(\cdot)$, etc.) that gives rise to parameter MP_{ijl}^{k,MS_α}, the meaning of which depends on the actual $\zeta(\cdot)$ used (see expressions below). Finally, D_{ij}^{k,MS_α} is the length of path $\Gamma_{ij}^{k,MS_\alpha}$, which is obtained by summing the length $\gamma_{ijl}^{k,MS_\alpha}$ of each of its segments.

Now, let $\chi_k(\cdot)$ denote an indicator function (i.e., $\chi_k(\mathbf{x}) = 1$ if $\mathbf{x}$ belongs to a water-filled region and $\chi_k(\mathbf{x}) = 0$ otherwise; then we define

$$MC_{ij}^{k,MS_\alpha} = \sum_{l=2}^{N_{ij}^{k,MS_\alpha}} \frac{\gamma_{ijl}^{k,MS_\alpha} MP_{ij}^{k,MS_\alpha}(M_k(\mathbf{x}))}{D_{ij}^{k,MS_\alpha}} \quad (26)$$

$$W_{ij}^{k,MS_\alpha} = \sum_{l=2}^{N_{ij}^{k,MS_\alpha}} \frac{\gamma_{ijl}^{k,MS_\alpha} MP_{ij}^{k,MS_\alpha}(\chi_k(\mathbf{x}))}{D_{ij}^{k,MS_\alpha}} \tag{27}$$

$$ND_{ij}^{k,MS_\alpha} = \frac{\|\mathbf{y}_j - \mathbf{y}_i\|}{D_{ij}^{k,MS_\alpha}} \tag{28}$$

$$CD_{ij}^{k,MS_\alpha} = \sum_{l=1}^{N_{ij}^{k,MS_\alpha}} \frac{d_2(\Gamma_{ij}^{k,MS_\alpha}(l), \Gamma_{ji}^{k,MS_\alpha})}{N_{ij}^{k,MS_\alpha}} \tag{29}$$

$$<\eta>_{ij}^{k,MS_\alpha} = \frac{\eta_{ij}^{k,MS_\alpha} + \eta_{ji}^{k,MS_\alpha}}{2} \tag{30}$$

with MC_{ij}^{k,MS_α} the mean local cost along $\Gamma_{ij}^{k,MS_\alpha}$, W_{ij}^{k,MS_α} the fraction of the path length that traverses flooded areas or watercourses, ND_{ij}^{k,MS_α} the Euclidean distance between the path endpoints divided by the path length, $d_2(\mathbf{x}, \Gamma)$ the minimum Euclidean distance from $\mathbf{x}$ to Γ, CD_{ij}^{k,MS_α} the average $d_2(\Gamma_{ij}^{k,MS_\alpha}(l), \Gamma_{ji}^{k,MS_\alpha})$ of all forward path vertices to the backward path and $<\eta>_{ij}^{k,MS_\alpha}$ is the average of some parameter η (which may be D, MC, W, etc.) in the forward and backward paths. Both W_{ij}^{k,MS_α} and ND_{ij}^{k,MS_α} take on values within the interval $[0, 1]$, and are unitary in the ideal case; however, it often happens that increasing W_{ij}^{k,MS_α} comes at the cost of increasing the path length, thus lowering ND_{ij}^{k,MS_α}. Their values also depend heavily on the terrain configuration between path endpoints.

For each scheme and stencil set, $<MC>_{ij}^{k,MS_\alpha}$, $<W>_{ij}^{k,MS_\alpha}$, $<ND>_{ij}^{k,MS_\alpha}$ and $<CD>_{ij}^{k,MS_\alpha}$ were computed for every $(\mathbf{y}_i, \mathbf{y}_j, M_k(\mathbf{x}))$ in the test set.

In order to test whether MS_c generates a lower local cost than MS_1 and MS_2, and to remove variability due to the different local cost map configurations that the paths cross over, the mean local cost obtained by MS_c is subtracted from the ones achieved by MS_1 and MS_2conveys. In Fig. 10a, the boxplot of $<MC>_{ij}^{k,MS_1} - <MC>_{ij}^{k,MS_c}$ and $<MC>_{ij}^{k,MS_2} - <MC>_{ij}^{k,MS_c}$ is shown, and it may be observed that most values are positive, which implies that MS_c achieved a lower mean local cost than the original schemes. Applying a right tailed Wilcoxon signed rank (WSR) test to both sets, the alternate hypothesis (the median is greater than zero) was selected in both cases with $p < 10^{-20}$.

The previous approach to remove variability due to the terrain configuration is also followed for $<W>_{ij}^{k,MS_\alpha}$ and $<ND>_{ij}^{k,MS_\alpha}$, and thus, the value obtained by MS_c is subtracted from the ones yielded by MS_1 and MS_2. The boxplots of the results are given in Figs. 10b and 10c. A left tailed WSR test is applied to all sample sets. The alternate hypothesis (the median is less than zero) was chosen with $p < 10^{-28}$ in both $<W>_{ij}^{k,MS_1} - <W>_{ij}^{k,MS_c}$ and $<W>_{ij}^{k,MS_2} - <W>_{ij}^{k,MS_c}$. The alternate hypothesis was also selected for $<ND>_{ij}^{k,MS_1} - <ND>_{ij}^{k,MS_c}$ and $<ND>_{ij}^{k,MS_2} - <ND>_{ij}^{k,MS_c}$, with $p < 10^{-36}$ and $p = 0.0485$. These results point out that the paths computed using MS_c have a better balance between path length and coverage of flooded terrain.

The boxplots for $<CD>_{ij}^{k,MS_\alpha}$, the mean Euclidean distance between forward and backward paths, are shown in

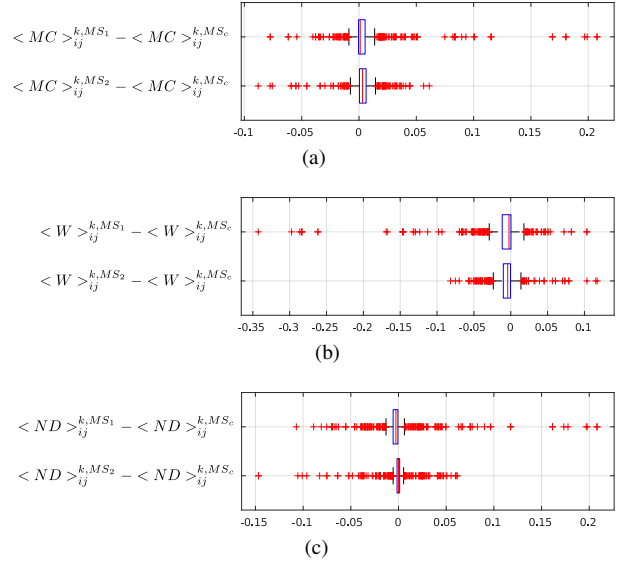Figure 10. Boxplot of the differences between MS_1 and MS_c, and between MS_2 and MS_c, of (a) the mean local cost $<MC>$, (b) the ratio of path length covered by water $<W>$, and (c) $<ND>$.
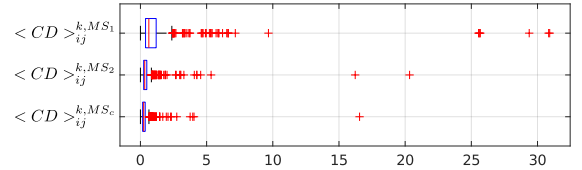

Figure 11. Boxplots of the mean Euclidean distance in pixels between forward and backward paths using MS_1, MS_2 and MS_c.

Fig. 11. It may be observed that the MS_c scheme attains lower values than the original schemes, which entails that the forward and backward paths are more similar to the proposed method. The distance between paths yielded by MS_c was subtracted from the results with MS_1 and MS_2, and a right-tailed WSR test was applied to both sets. The alternate hypothesis was selected with $p < 10^{-26}$ in both cases, that is, the median of $<CD>_{ij}^{k,MS_c}$ is lower than $<CD>_{ij}^{k,MS_1}$ and $<CD>_{ij}^{k,MS_2}$.

VI. Conclusions

In this paper we have proposed a new second order numerical scheme for MSFM, which improves the accuracy when the local cost is not constant; to this end, the MSFM method proposed in [13] has been thoroughly reviewed and a numerical scheme formulation for arbitrary stencils and regular anisotropic grids has also been provided.

The experimental results point out that using the 3-D 6-stencil set proposed in the original MSFM method may lead to perceptibly different outcomes if the stencil vector coordinates are permuted. This may be of importance in any imaging application in which the relative position of the imaged object with respect to the axes is not known beforehand, which is the case, for instance, in 3D cardiac magnetic resonance imaging [12]. To avoid this limitation, the use of permutation invariant stencil sets has been proposed. In addition, the orthogonal or nearly-orthogonal stencil sets under evaluation

have yielded better accuracy results than using a set of highly non-orthogonal stencils.

A case of use of the MSFM method for optimal path planning in UAV flood monitoring missions showed that paths computed using the proposed scheme had better tradeoff between coverage and path length than the ones computed with the first and second order original MSFM schemes.

The design of the stencil set is an important factor in the performance of the MSFM method. The question of which should be the maximum radius of the stencil set has been raised by the accuracy results on the sinusoidal local costs. It is a topic, however, that remains open at the moment and merits further study. Another line of research could be to study the feasibility of applying the multi-stencil centered scheme on problems with anisotropic local costs. The case of local costs composed of symmetric second rank positive semidefinite tensors are promising, since in that case one could arrive to an equation formally equivalent to Eq. (14). However, it should be checked whether other requisites, such as the causality condition, still hold in this scenario.

APPENDIX

SECOND ORDER APPROXIMATION TO THE DERIVATIVES

This Appendix contains the proof that the proposed approximation to the directional derivatives are second order accurate. As in Section III-A, the exposition below is given for $D = 3$. The Taylor expansion of $T(\mathbf{z})$ on $\mathbf{x}_c$ is expressed as:

$$
\begin{aligned}
T(\mathbf{z}) = {} & T(\mathbf{x}_c) + (\mathbf{z} - \mathbf{x}_c)^T \nabla T(\mathbf{x}_c) \\
& + \frac{1}{2}(\mathbf{z} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\mathbf{z} - \mathbf{x}_c) \\
& + \sum_{|\mathbf{n}|=3}^{\infty} \frac{\partial^{\mathbf{n}} T(\mathbf{x}_c)}{\mathbf{n}!}(\mathbf{z} - \mathbf{x}_c)^{\mathbf{n}}
\end{aligned}
\tag{31}
$$

where the multi-index notation[3] has been employed for the terms of order 3 and higher. Plugging Eq. (31) into the scheme given in Eq. (9), we have:

$$
\begin{aligned}
2^{D-1}\|\mathbf{v}_j\|\widehat{D}^{-\mathbf{v}_j}T(\mathbf{x}) = {} & \sum_{\hat{\mathbf{z}} \in \mathcal{N}_j^0} T(\hat{\mathbf{z}}) - \sum_{\mathbf{z} \in \mathcal{N}_j^-} T(\mathbf{z}) \\
= {} & \sum_{\hat{\mathbf{z}} \in \mathcal{N}_j^0} (\hat{\mathbf{z}} - \mathbf{x}_c)^T \nabla T(\mathbf{x}_c) - \sum_{\mathbf{z} \in \mathcal{N}_j^-} (\mathbf{z} - \mathbf{x}_c)^T \nabla T(\mathbf{x}_c) \\
& + \frac{1}{2} \sum_{\hat{\mathbf{z}} \in \mathcal{N}_j^0} (\hat{\mathbf{z}} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\hat{\mathbf{z}} - \mathbf{x}_c) \\
& - \frac{1}{2} \sum_{\mathbf{z} \in \mathcal{N}_j^-} (\mathbf{z} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\mathbf{z} - \mathbf{x}_c) \\
& + \sum_{|\mathbf{n}|=3}^{\infty} \frac{\partial^{\mathbf{n}} T(\mathbf{x}_c)}{\mathbf{n}!}\left(\sum_{\hat{\mathbf{z}} \in \mathcal{N}_j^0} (\hat{\mathbf{z}} - \mathbf{x}_c)^{\mathbf{n}} - \sum_{\mathbf{z} \in \mathcal{N}_j^-} (\mathbf{z} - \mathbf{x}_c)^{\mathbf{n}}\right)
\end{aligned}
\tag{32}
$$

[3] The multi-index notation employed means that $\mathbf{n} = (n_1, n_2, n_3)$, $|\mathbf{n}| = n_1 + n_2 + n_3$, $\mathbf{n}! = n_1! n_2! n_3!$, $\mathbf{x}^{\mathbf{n}} = x_1^{n_1} x_2^{n_2} x_3^{n_3}$, and $\partial^{\mathbf{n}} = \frac{\partial^{n_1+n_2+n_3}}{\partial x_1^{n_1} \partial x_2^{n_2} \partial x_3^{n_3}}$.

where the stencil superindex has been dropped, in order to simplify the notation. Assume, without loss of generality, that $\mathbf{v}_j = \mathbf{v}_3$. Since $\mathbf{x}_c$ is the geometric center of the parallelepiped vertices, $\mathbf{x}_c = \mathbf{x} + \frac{1}{2}(\mathbf{v}_1 + \mathbf{v}_2 - \mathbf{v}_3)$. Then,

$$
\begin{aligned}
\sum_{\mathbf{z} \in \mathcal{N}_3^-} (\mathbf{z} - \mathbf{x}_c) = {} & \sum_{\mathbf{z} \in \mathcal{N}_3^-} \mathbf{z} - 4\mathbf{x}_c = \sum_{s_1, s_2=0}^{1} \left(\mathbf{x} + s_1 \mathbf{v}_1\right. \\
& \left. + s_2 \mathbf{v}_2 - \mathbf{v}_3\right) - 4\left(\mathbf{x} + \frac{\mathbf{v}_1 + \mathbf{v}_2 - \mathbf{v}_3}{2}\right) \\
= {} & -4\mathbf{v}_3 + 2\mathbf{v}_3 = -2\mathbf{v}_3
\end{aligned}
\tag{33}
$$

Likewise, for $\mathcal{N}_3^0$, we have $\sum_{\hat{\mathbf{z}} \in \mathcal{N}_3^0} (\hat{\mathbf{z}} - \mathbf{x}_c) = 2\mathbf{v}_3$. The same procedure can be applied to $j = 1$ and $j = 2$ as well. For the second order terms,

$$
\begin{aligned}
& \frac{1}{2} \sum_{\mathbf{z} \in \mathcal{N}_3^-} (\mathbf{z} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\mathbf{z} - \mathbf{x}_c) \\
& = \frac{1}{2} \sum_{\mathbf{z} \in \mathcal{N}_3^-} \sum_{l=1}^{D} \sum_{m=1}^{D} (z_l - x_{c,l}) \frac{\partial^2 T(\mathbf{x}_c)}{\partial x_l \partial x_m}(z_m - x_{c,m}) \\
& = \frac{1}{2} \sum_{l=1}^{D} \sum_{m=1}^{D} \frac{\partial^2 T(\mathbf{x}_c)}{\partial x_l \partial x_m} \sum_{\mathbf{z} \in \mathcal{N}_3^-} (z_l - x_{c,l})(z_m - x_{c,m})
\end{aligned}
\tag{34}
$$

where:

$$
\begin{aligned}
& \sum_{\mathbf{z} \in \mathcal{N}_3^-} (z_l - x_{c,l})(z_m - x_{c,m}) \\
& = \sum_{s_1, s_2=0}^{1} \left(s_1 v_{1,l} + s_2 v_{2,l} - v_{3,l} - \frac{v_{1,l} + v_{2,l} - v_{3,l}}{2}\right) \\
& \quad \cdot \left(s_1 v_{1,m} + s_2 v_{2,m} - v_{3,m} - \frac{v_{1,m} + v_{2,m} - v_{3,m}}{2}\right) \\
& = \sum_{s_1, s_2=\pm\frac{1}{2}} \left(s_1 v_{1,l} + s_2 v_{2,l} - \frac{v_{3,l}}{2}\right) \\
& \quad \cdot \left(s_1 v_{1,m} + s_2 v_{2,m} - \frac{v_{3,m}}{2}\right)
\end{aligned}
\tag{35}
$$

Taking into account that $\sum_{s=\pm\frac{1}{2}} s\xi = 0$ and, consequently, $\sum_{s_1=\pm\frac{1}{2}} \sum_{s_2=\pm\frac{1}{2}} s_1 s_2 \xi = 0$, then

$$
\begin{aligned}
& \sum_{\mathbf{z} \in \mathcal{N}_3^-} (z_l - x_{c,l})(z_m - x_{c,m}) \\
& = \sum_{s_1, s_2=\pm\frac{1}{2}} s_1^2 v_{1,l} v_{1,m} + s_2^2 v_{2,l} v_{2,m} + \frac{v_{3,l} v_{3,m}}{4} \\
& = v_{1,l} v_{1,m} + v_{2,l} v_{2,m} + v_{3,l} v_{3,m}
\end{aligned}
\tag{36}
$$

Repeating for the other parallelepiped face, $\mathcal{N}_3^0$, we have

that:

$$\frac{1}{2} \sum_{\hat{\mathbf{z}} \in \mathcal{N}_3^0} (\hat{\mathbf{z}} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\hat{\mathbf{z}} - \mathbf{x}_c)$$

$$= \frac{1}{2} \sum_{l=1}^{D} \sum_{m=1}^{D} \frac{\partial^2 T(\mathbf{x}_c)}{\partial x_l \partial x_m} (v_{1,l} v_{1,m} + v_{2,l} v_{2,m} + v_{3,l} v_{3,m}) \quad (37)$$

$$= \frac{1}{2} \sum_{\hat{\mathbf{z}} \in \mathcal{N}_3^-} (\hat{\mathbf{z}} - \mathbf{x}_c)^T \nabla^2 T(\mathbf{x}_c)(\hat{\mathbf{z}} - \mathbf{x}_c)$$

Thus, the terms with second derivatives of one face cancel out those of the other face, and Eq. (32) becomes:

$$4\|\mathbf{v}_j\|\widehat{D}^{-\mathbf{v}_j} T(\mathbf{x}) = (2\mathbf{v}_j)^T \nabla T(\mathbf{x}_c) - (-2\mathbf{v}_j)^T \nabla T(\mathbf{x}_c)$$
$$+ \frac{1}{2}(2\mathbf{v}_j)^T \nabla^2 T(\mathbf{x}_c)(2\mathbf{v}_j) - \frac{1}{2}(-2\mathbf{v}_j)^T \nabla^2 T(\mathbf{x}_c)(-2\mathbf{v}_j)$$
$$+ R(\mathbf{x}) = 4(\mathbf{v}_j)^T \nabla T(\mathbf{x}_c) + R(\mathbf{x})$$
$$= 4\|\mathbf{v}_j\| \left\langle \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|}, \nabla T(\mathbf{x}_c) \right\rangle + R(\mathbf{x})$$

$$(38)$$

with $R(\mathbf{x})$ the error arising from the truncation of the Taylor series, which is $O(\max_k(\|\mathbf{v}_k\|^3))$. Then, we have:

$$\widehat{D}^{-\mathbf{v}_j} T(\mathbf{x}) = \psi_j + \frac{R(\mathbf{x})}{4\|\mathbf{v}_j\|} = \psi_j + O(\max_k(\|\mathbf{v}_k\|^2)) \quad (39)$$

REFERENCES

[1] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proc. Natl. Acad. Sci.*, vol. 93, no. 4, pp. 1591–1595, feb 1996.

[2] T. Deschamps and L. D. Cohen, "Fast extraction of minimal paths in 3D images and applications to virtual endoscopy," *Med. Image Anal.*, vol. 5, no. 4, pp. 281–299, 2001.

[3] M. S. Hassouna and A. A. Farag, "Variational curve skeletons using gradient vector flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2257–2274, 2009.

[4] R. Kimmel and J. A. Sethian, "Optimal algorithm for shape from shading and path planning," *J. Math. Imaging Vision*, vol. 14, no. 3, pp. 237–244, 2001.

[5] S. Y. Yuen, Y. Y. Tsui, and C. K. Chow, "A fast marching formulation of perspective shape from shading under frontal illumination," *Pattern Recognit. Lett.*, vol. 28, no. 7, pp. 806–824, 2007.

[6] A. Telea, "An image inpainting technique based on the fast marching method," *J. Graph. Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[7] J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science.* Cambridge, United Kingdom: Cambridge University Press, 1999.

[8] M. Lynch, O. Ghita, and P. F. Whelan, "Left-ventricle myocardium segmentation using a coupled level-set with a priori knowledge," *Comput. Med. Imaging Graph.*, vol. 30, no. 4, pp. 255–262, 2006.

[9] J. Yan and T. Zhuang, "Applying improved fast marching method to endocardial boundary detection in echocardiographic images," *Pattern Recognit. Lett.*, vol. 24, no. 15, pp. 2777–2784, 2003.

[10] C. Lenglet, M. Rousson, R. Deriche, and O. Faugeras, "Statistics on the manifold of multivariate normal distributions: Theory and application to diffusion tensor MRI processing," *J. Math. Imaging Vision*, vol. V25, no. 3, pp. 423–444, 2006.

[11] M. Schwenke, A. Hennemuth, B. Fischer, and O. Friman, "A novel anisotropic fast marching method and its application to blood flow computation in phase-contrast MRI," *Methods Inform. Med.*, vol. 51, no. 5, pp. 423–428, 2012.

[12] S. Merino-Caviedes, L. Cordero-Grande, A. Revilla-Orodea, T. Sevilla-Ruiz, M. T. Pérez, M. Martín-Fernández, and C. Alberola-López, "Multi-stencil streamline fast marching: a general 3D framework to determine myocardial thickness and transmurality in late enhancement images," *IEEE Trans. Med. Imaging*, vol. 33, no. 1, pp. 23–37, 2014.

[13] M. S. Hassouna and A. A. Farag, "Multistencils fast marching methods: A highly accurate solution to the Eikonal equation on cartesian domains," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1563–1574, 2007.

[14] J. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Trans. Automat. Contr.*, vol. 40, no. 9, pp. 1528–1538, 1995.

[15] P.-E. Danielsson and Q. Lin, "A modified fast marching method," in *Proceedings of the 13th Scandinavian Conference on Image Analysis*, ser. SCIA'03. Berlin, Heidelberg: Springer-Verlag, 2003, pp. 1154–1161.

[16] V. Appia and A. Yezzi, "Symmetric fast marching schemes for better numerical isotropy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 99, pp. 2298–2304, 2013.

[17] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, 1959.

[18] D. Monniaux, "The pitfalls of verifying floating-point computations," *ACM Trans. Program. Lang. Syst.*, vol. 30, no. 3, p. 12, 2008.

[19] Mathworks. Fixed-point designer. Visited on November 2014. [Online]. Available: http://www.mathworks.com/help/fixedpoint/

[20] National Aeronautics and Space Administration, "NRT Global Flood Mapping." [Online]. Available: https://floodmap.modaps.eosdis.nasa.gov/

[21] R. Ardon and L. D. Cohen, "Fast constrained surface extraction by minimal paths," *Int. J. Comput. Vis.*, vol. 69, no. 1, pp. 127–136, 2006.