Level Set Method for Surface Reconstruction and Its Application in Surveying

Jana Haličková, Ph.D.[1]; and Karol Mikula, Ph.D.[2]

Abstract: A novel approach to a three-dimensional (3D) model creation from data sets obtained by laser scanning or photogrammetry is presented, including a description of the method and efficient numerical algorithms for 3D model creation for surveying applications. The presented method solves the fundamental problem of surface reconstruction by constructing models using differential geometry and partial differential equations. The time-relaxed eikonal equation was used to obtain the distance function from the cloud of points, and the 3D model was then created by the evolution of an initial guess by the advection equation regularized by curvature, where the advective velocity is given by the gradient of the computed distance function. The partial differential equations were discretized by the finite-difference schemes, and the results of the 3D model creation are presented and discussed. **DOI: 10.1061/(ASCE)SU.1943-5428.0000159.** *© 2016 American Society of Civil Engineers.*

Introduction

The creation of a three-dimensional (3D) model from a data set obtained by laser scanning is often a very complicated and time-consuming process. The principal aim of processing the data is the creation of the model (S), which approximates the real shape (S) as much as possible. The main difficulties of surface reconstruction from point clouds include (1) an unknown connectivity or ordering information among the data points, (2) an unknown topology of the original surface, and (3) a noise and nonuniformity of the data.

There are many approaches to the processing of these data sets. Commercial software mostly uses three of them: triangulated surfaces, nonuniform B-spline surfaces (NURBS), and the replacement of data points by mathematically defined objects. The first two methods are used for creating models of objects with complicated topology, and the third one is used when it is possible to replace some parts of the object by mathematically defined entities, such as a sphere, cube, block, cylinder, or spire. Different software packages for different tasks of model creation are available. However, because of the high price of these packages, it is sometimes difficult and expensive to use the best one for each task. Because laser scanning is widely applicable in various fields, efficient processing algorithms have also been developed for particular tasks (Dorninger and Pfeifer 2008; El Meouche et al. 2013).

This paper presents a new approach to 3D model creation based on the level set method (LSM). The LSM solves the level set equations in which the evolution of the level set function is given by solution of the partial differential equations (PDEs). The use of LSM is diverse (i.e., there are many applications in physics, material science, image processing, or computer graphics; Bourgine et al. 2009; Osher and Fedkiw 2002; Osher and Paragios 2003; Sethian 1999). This paper uses LSM for the model creation in surveying where the unorganized set of points is obtained by laser or optical scanning or photogrammetry. This method is called level set method for surface reconstruction (LSMSR), and it consists of the solution of two PDEs. Solution of the first PDE gives the distance function to point cloud, whereas the second one gives the function that represents the final 3D model. The whole solution is performed on a simple rectangular grid in a chosen computational domain (Ω). The set of points (Ω_0) obtained by 3D scanning or photogrammetric methods is a subset of Ω. The LSM is used as a numerical tool to deform an initial guess to its final form. The 3D model is created as an isosurface of the final level set function given on a fixed rectangular grid. This work is based on an idea presented by Zhao et al. (2000, 2001), but opposite to that approach, the authors do not follow just one isosurface of the evolving level set function, as it is standard in the classical LSM. Instead, an idea of the subjective surface method is used (Sarti et al. 2000), and the final 3D model is detected as one of the isosurfaces of a shock-like profile in the final shape of the level set function (see also Corsaro et al. 2006; Mikula and Sarti 2007). This makes the method robust with respect to the isosurface choice and represents the novelty in comparison to Zhao et al. (2000, 2001).

The paper is organized as follows: "Mathematical Formulation of LSMSR" describes the formulation of the authors' approach. "Numerical Solution of LSMSR" proposes a numerical solution to this mathematical model. In "Quantitative Tests of the Algorithm" the proposed procedure is tested on the specific representative examples, and in "Practical Examples of 3D Model Creation Using LSMSR," some real examples of model creation from data obtained by 3D scanning and photogrammetry are presented.

Mathematical Formulation of LSMSR

As already mentioned in the introduction, this method of the model creation is based on the solution of two PDEs. The distance function to the 3D point cloud is given as the stationary solution of the so-called time-relaxed eikonal equation

$$d_t + |\nabla d| = 1 \tag{1}$$

coupled with a Dirichlet-type condition

[1] Researcher, Monument Board of the Slovak Republic, Cesta na Červený most 6, 81406 Bratislava, Slovakia. E-mail: jana.halickova@gmail.com

[2] Professor, Dept. of Mathematics and Descriptive Geometry, Faculty of Civil Engineering, Slovak Univ. of Technology, Radlinského 11, 81005 Bratislava, Slovakia (corresponding author). E-mail: karol.mikula@stuba.sk

Note. This manuscript was submitted on December 30, 2013; approved on September 28, 2015; published online on February 8, 2016. Discussion period open until July 8, 2016; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Surveying Engineering***, © ASCE, ISSN 0733-9453.

$$d(x) = 0 \quad \text{for} \quad x \in \Omega_0 \subset \Omega \qquad (2)$$

where Ω = computational domain; and Ω_0 = set of measured point clouds. The second partial differential equation is the advection equation with the curvature term

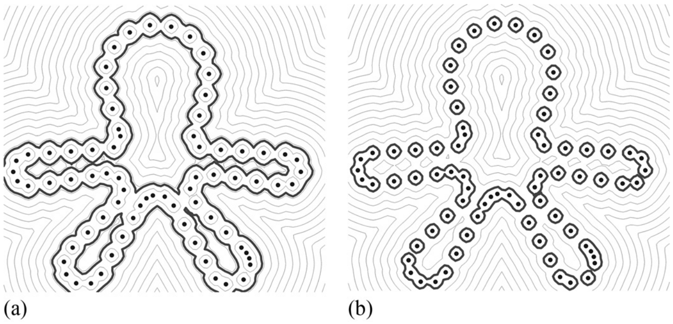

Fig. 1. Influence of parameter β on the shape of the initial guess: (a) right choice; (b) a wrong choice

$$u_t - \nabla d \cdot \nabla u + \delta |\nabla u| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) = 0 \qquad (3)$$

where $(x, t) \in \Omega \times [0, T_S]$ and $v = -\nabla d$ = advective velocity given by the gradient of the computed distance function taken with the minus sign. The parameter $\delta \in [0, 1]$ in the last term determines the influence of curvature on the final shape of the created object. Eq. (3) is accompanied by the Dirichlet boundary conditions and an initial condition that will be discussed in the section "Finding an Initial Surface."

Numerical Solution of LSMSR

The numerical solution for this mathematical model [Eqs. (1)–(3)] consists of three steps:

- finding a distance function to the point cloud by solving Eqs. (1) and (2),
- finding an initial surface for the evolution Eq. (3), and
- generating a final model by solving the level set Eq. (3).

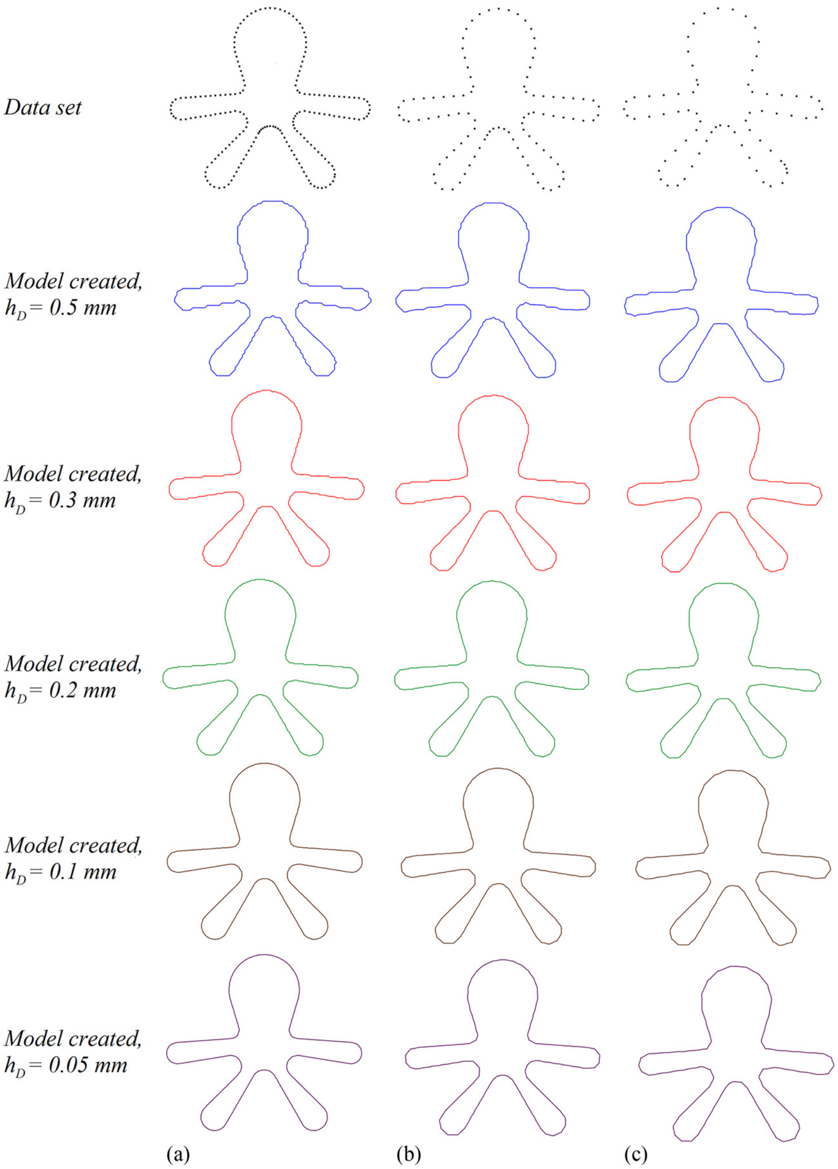
Data set
Model created, h_D= 0.5 mm
Model created, h_D= 0.3 mm
Model created, h_D= 0.2 mm
Model created, h_D= 0.1 mm
Model created, h_D= 0.05 mm
(a)
(b)
(c)

Fig. 2. Object models created for different values of h_D from the input data set with point densities of (a) 1 mm; (b) 2 mm; and (c) 3 mm

Table 1. Values of *HD(A,B)* for Data Sets ch1, ch2, and ch3, Depending on the Mesh Density Parameter (h_D)

	Grid size h_D									
HD(A,B)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
HD(A,B) for input points ch1 with density 1 mm	**0.0154**	0.0279	0.0445	0.0643	0.0741	0.0883	0.1070	0.1235	0.1396	0.1504
HD(A,B) for input points ch2 with density 2 mm	**0.0161**	0.0296	0.0453	0.0689	0.0757	0.0846	0.1133	0.1196	0.1395	0.1458
HD(A,B) for input points ch3 with density 3 mm	**0.0153**	0.0297	0.0441	0.0676	0.0716	0.0902	0.1105	0.1224	0.1465	0.1509

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 2. Values of *HD(B,A)* for Data Sets ch1, ch2, and ch3, Depending on the Mesh Density Parameter (h_D)

	Grid size h_D									
HD(B,A)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
HD(B,A) for input points ch1 with density 1 mm	0.2427	**0.2420**	**0.2427**	**0.2420**	**0.2421**	0.2452	0.2500	0.2527	0.2579	0.2704
HD(B,A) for input points ch2 with density 2 mm	0.4740	0.4747	0.4730	**0.4707**	**0.4717**	**0.4707**	0.4742	0.4725	0.4749	0.4798
HD(B,A) for input points ch3 with density 3 mm	0.6598	0.6621	0.6615	0.6586	**0.6575**	0.6586	0.6603	0.6595	0.6678	0.6651

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

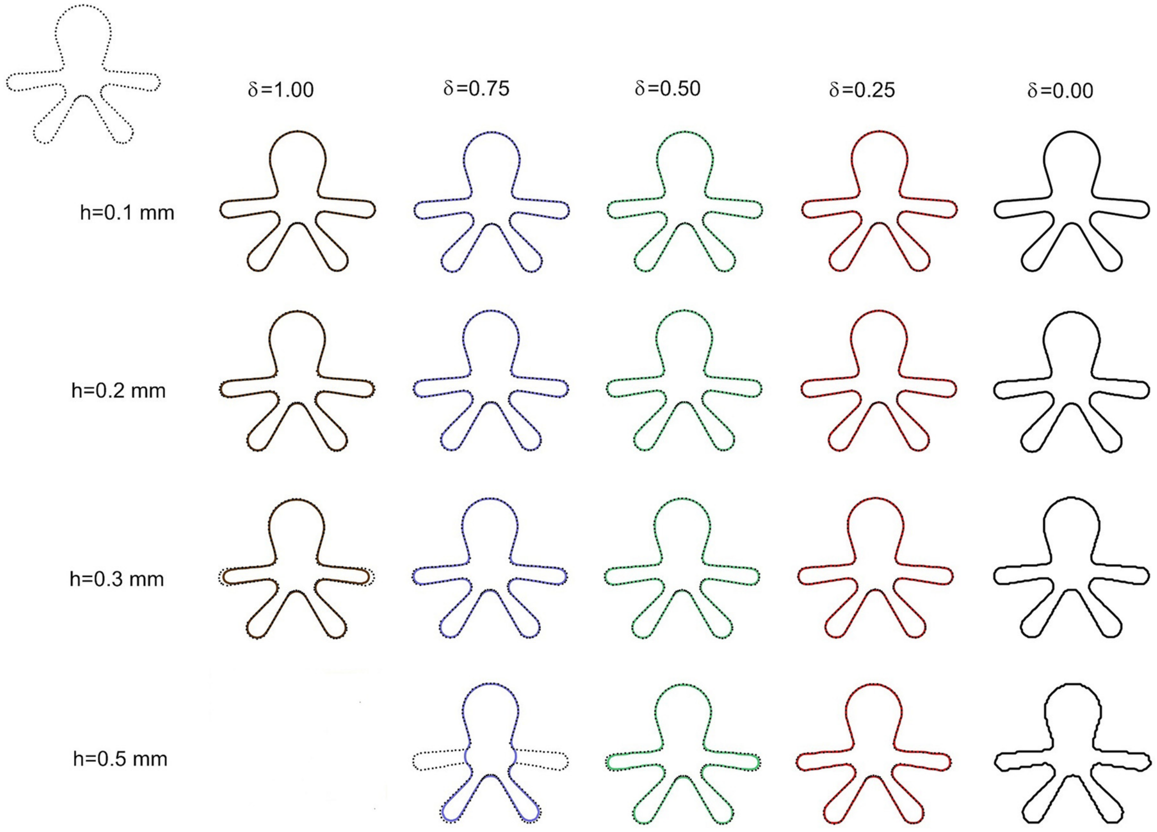

Fig. 3. Influence of parameter δ on the final model for the data set with point density 1 mm

Finding a Distance Function

For the numerical solution of the time-relaxed eikonal Eq. (1) with the condition Eq. (2), the relaxation method with fixing described by Bourgine et al. (2009) is used. The method is simply implemented and efficient for 3D computations. It is based on explicit time discretization with time step (τ_D) and on the spatial discretization based on the Rouy-Tourin scheme (Rouy and Tourin 1992). A discretization of the computational domain Ω in the form of a uniform 3D grid of cubic elements with the edge size h_D is assumed, and the element of such grid is called a voxel. The choice of parameter h_D depends on the density of the measured points. A position of every grid point is defined by three indices (i, j, k). After discretization of Ω, the exact distance is assigned to the grid points that are close to the measured data points. These values are fixed in further computations. A small positive value is assigned to all other grid points in the initialization step. The values of these nonfixed grid points are subsequently updated by the following algorithm.

Let $d^n_{i,j,k}$ represent the value of the numerical solution of Eqs. (1)–(2) in a grid point i, j, k at time step n. The value at new time step $n+1$ in any nonfixed point is given by

Table 3. Values of *HD(A,B)* for Data Set ch3 Depending on the Mesh Density Parameter (h_D) and the Curvature Parameter (δ)

	Parameter δ								
HD(A,B) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(A,B) for $h_D = 0.5$	0.1509	0.1460	**0.1448**	0.1470	0.1487	0.1519	0.1561	0.1601	0.1641
HD(A,B) for $h_D = 0.4$	0.1238	0.1174	**0.1158**	0.1173	0.1198	0.1230	0.1263	0.1298	0.1335
HD(A,B) for $h_D = 0.3$	0.0902	**0.0879**	**0.0880**	0.0893	0.0920	0.0946	0.0974	0.1003	0.1033

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 4. Values of *HD(B,A)* for Data Set ch3 Depending on the Mesh Density Parameter (h_D) and the curvature parameter (δ)

	Parameter δ								
HD(A,B) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(B,A) for $h_D = 0.5$	0.6651	**0.6638**	**0.6637**	**0.6639**	0.6644	0.6654	0.6695	0.6703	0.6712
HD(B,A) for $h_D = 0.4$	0.6603	0.6582	**0.6571**	0.6585	0.6584	0.6586	0.6589	0.6615	0.6621
HD(B,A) for $h_D = 0.3$	**0.6586**	**0.6588**	0.6599	0.6600	0.6634	0.6639	0.6645	0.6652	0.6658

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 5. Values of *HD(A,B)* for Data Set ch2 Depending on the Mesh Density Parameter (h_D) and the Curvature Parameter (δ)

	Parameter δ								
HD(A,B) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(A,B) for $h_D = 0.4$	0.1204	0.1119	**0.1113**	0.1133	0.1158	0.1181	0.1204	0.1224	0.1246
HD(A,B) for $h_D = 0.3$	0.0846	0.0806	**0.0797**	0.0809	0.0829	0.0851	0.0875	0.0904	0.0936
HD(A,B) for $h_D = 0.2$	0.0689	**0.0643**	**0.0641**	0.0649	0.0663	0.0680	0.0698	0.0715	0.0730

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 6. Values of *HD(B,A)* for Data Set ch2 Depending on the Mesh Density Parameter (h_D) and the Curvature Parameter (δ)

	Parameter δ								
HD(B,A) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(B,A) for $h_D = 0.4$	0.4731	**0.4699**	**0.4704**	0.4733	0.4731	0.4745	0.4751	0.4759	0.4783
HD(B,A) for $h_D = 0.3$	0.4707	**0.4694**	**0.4695**	**0.4695**	0.4698	0.4702	0.4707	0.4713	0.4730
HD(B,A) for $h_D = 0.2$	**0.4707**	**0.4707**	**0.4706**	**0.4706**	**0.4707**	0.4720	0.4722	0.4731	0.4734

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 7. Values of *HD(A,B)* for Data Set ch1 Depending on the Mesh Density Parameter (h_D) and the Curvature Parameter (δ)

	Parameter δ								
HD(A,B) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(A,B) for $h_D = 0.3$	0.0883	0.0839	**0.0821**	**0.0819**	0.0824	0.0835	0.0850	0.0865	0.0880
HD(A,B) for $h_D = 0.2$	0.0643	**0.0606**	**0.0604**	0.0608	0.0615	0.0622	0.0633	0.0647	0.0661
HD(A,B) for $h_D = 0.1$	0.0279	**0.0242**	**0.0243**	0.0247	0.0253	0.0259	0.0265	0.0273	0.0281

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

Table 8. Values of *HD(B,A)* for Data Set ch1 Depending on the Mesh Density Parameter (h_D) and the Curvature Parameter δ

	Parameter δ								
HD(A,B) (mm)	0.000	0.025	0.050	0.075	0.100	0.125	0.150	0.175	0.200
HD(B,A) for $h_D = 0.3$	0.2452	0.2427	**0.2421**	**0.2421**	0.2425	0.2430	0.2436	0.2444	0.2453
HD(B,A) for $h_D = 0.2$	0.2420	**0.2410**	**0.2414**	0.2419	0.2423	0.2426	0.2430	0.2434	0.2443
HD(B,A) for $h_D = 0.1$	**0.2420**	**0.2418**	**0.2418**	0.2421	0.2424	0.2427	0.2429	0.2431	0.2433

Note: Bold values indicate the optimal (smallest) value or a range of optimal values in a row.

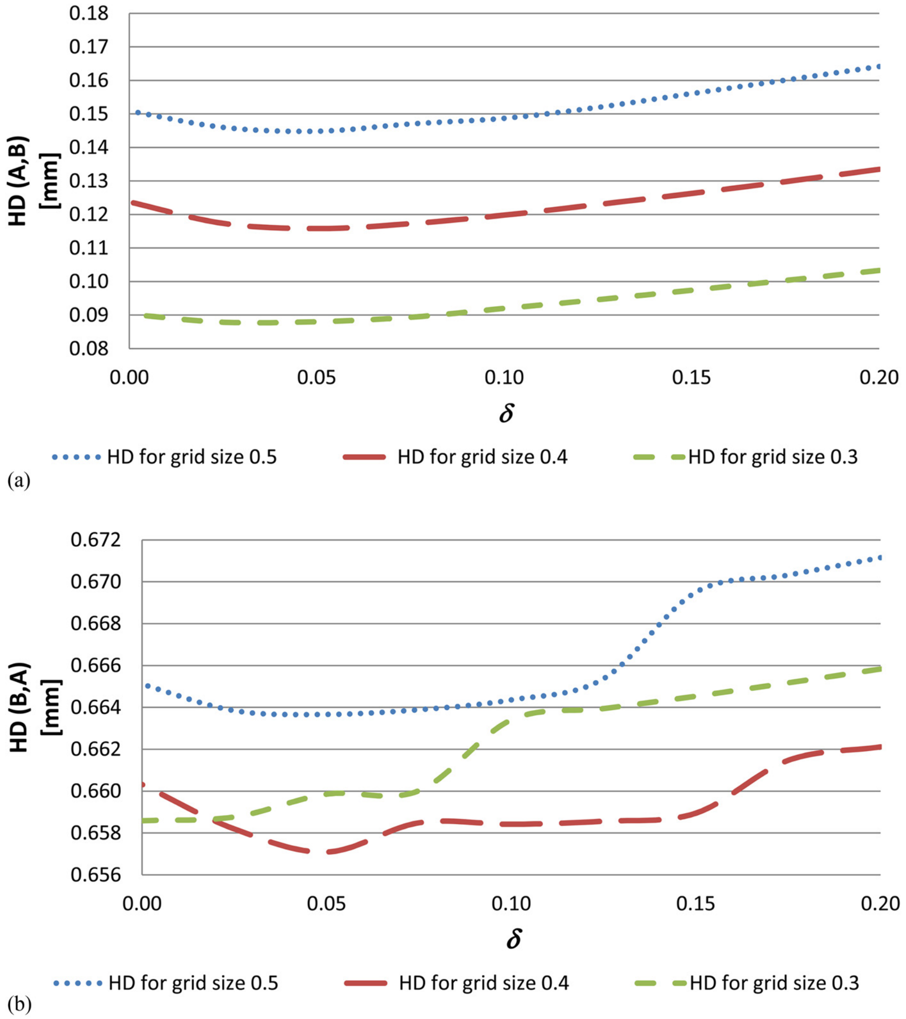

Fig. 4. Plots of (a) *HD(A,B)* and (b) *HD(B,A)* for the data set ch3 depending on the mesh density parameter h_D and the curvature parameter δ

$$d_{i,j,k}^{n+1} = d_{i,j,k}^{n} + \tau_D - \frac{\tau_D}{h_D}\sqrt{\max(M_{i,j,k}^{-1,0,0},\ M_{i,j,k}^{1,0,0}) + \max(M_{i,j,k}^{0,-1,0},\ M_{i,j,k}^{0,1,0}) + \max(M_{i,j,k}^{0,0,-1},\ M_{i,j,k}^{0,0,1})} \tag{4}$$

where $M_{i,j,k}^{p,q,r} = (\min(d_{i+p,j+q,k+r}^{n} - d_{i,j,k}^{n}, 0))^2,\ p,q,r \in \{-1,0,1\}$, $|p| + |q| + |r| = 1$. Eq. (4) is stable for $\tau_D \leq h_D/2$ and gives values gradually approaching the approximate distance function. This property allows one to utilize a fixing strategy. An index set (F) is introduced that contains the indices $\{i,j,k\}$ of grid points, which already reached the steady state (i.e., difference between $d_{i,j,k}^{n+1}$ and $d_{i,j,k}^{n}$ is smaller than a prescribed tolerance. Such points are fixed and removed from further computations. The distance function computation is finished when all grid points are fixed.

Finding an Initial Surface

Because an initial surface is deformed to the final shape by solving the level set Eq. (3), it is very important to find a good initial guess [i.e., the initial condition (u_0) for the evolution Eq. (3)]. Although any initial surface that contains the measured data set can be used, the suitable choice of the initial condition is crucial for the efficiency of LSMSR.

To find the proper initial guess, the following tagging algorithm is used. First, an initial exterior region is created, which should be a subset of the true exterior domain. It contains voxels outside of the measured data points (e.g., only the voxels along the borders of the computational domain Ω). The set of all of these exterior voxels is denoted by F. A parameter (β) representing a certain value of the distance function is chosen. Then, a first voxel in the set F is taken, and whether it has a nonexterior neighbor with a value of distance function greater than β is determined. If such a neighbor exists, this point is tagged as exterior and added to the F. Then, a next exterior point from the set F is taken, and the procedure is repeated. The tagging algorithm is applied to any point only once, and it is finished when the last exterior point in the set F is checked.

The right choice of parameter β is very important. In the continuous limit, if $d(x)$ is the distance function to a smooth surface, the zero level set [i.e., the set $\{x:\ d(x) = \beta = 0\}$] is the true surface. From this point of view, one would like to choose a value of β as small as possible. However, in the point clouds discrete case, if a β value is too small, the contour may consist of separated small spheres around data points. Fig. 1 presents an example of the right and wrong choice of β. On the left, the contour $d(x) = \beta$ (thicker line) is the continuous line, and on the right, the contour consists of more discontinuous objects. The correct choice of β also depends on the sampling density of data.

Generating the Final Model

Numerical Solution of Level Set Equation without Curvature Term

If one does not want to include the influence of curvature on the final object shape, it is possible to use only the advective part of Eq. (3)

$$u_t - \nabla d \cdot \nabla u = 0 \tag{5}$$

Then, the explicit time discretization with the time step (τ_S) and the space discretization based on the upwind principle can be used. If the spatial grid size is h_D, and the central differences for the function d in coordinate directions $D^x_{i,j,k}d$, $D^y_{i,j,k}d$, and $D^z_{i,j,k}d$ are defined as

$$\begin{aligned} D^x_{i,j,k}d &= (d_{i+1,j,k} - d_{i-1,j,k})/(2h_D) \\ D^y_{i,j,k}d &= (d_{i,j+1,k} - d_{i,j-1,k})/(2h_D) \\ D^z_{i,j,k}d &= (d_{i,j,k+1} - d_{i,j,k-1})/(2h_D) \end{aligned} \tag{6}$$

then the numerical approximation of Eq. (5) is obtained in the form

$$u^{n+1}_{i,j,k} = u^n_{i,j,k} - \frac{\tau_S}{h_D}\left[\begin{array}{l} \max(-D^x_{i,j,k}d,\,0)\,(u^n_{i,j,k} - u^n_{i-1,j,k}) + \min(-D^x_{i,j,k}d,\,0)\,(u^n_{i+1,j,k} - u^n_{i,j,k}) + \\ + \max(-D^y_{i,j,k}d,\,0)\,(u^n_{i,j,k} - u^n_{i,j-1,k}) + \min(-D^y_{i,j,k}d,\,0)\,(u^n_{i,j+1,k} - u^n_{i,j,k}) + \\ + \max(-D^z_{i,j,k}d,\,0)\,(u^n_{i,j,k} - u^n_{i,k,k-1}) + \min(-D^z_{i,j,k}d,\,0)\,(u^n_{i,j,k+1} - u^n_{i,j,k}) \end{array}\right] \tag{7}$$

which is stable for $\tau_S \leq h_D/2$.

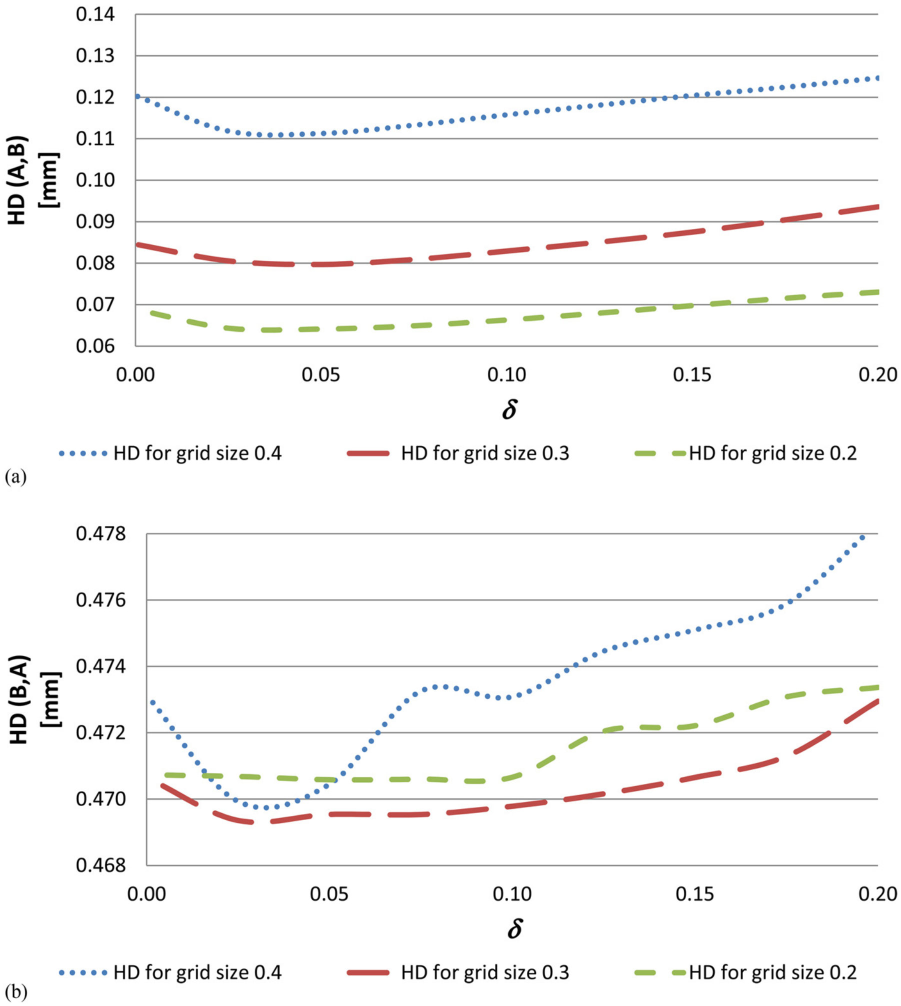

Fig. 5. Plots of (a) *HD(A,B)* and (b) *HD(B,A)* for the data set ch2 depending on the mesh density parameter h_D and the curvature parameter δ

Numerical Solution of Level Set Equation Including Curvature Term

As is shown in the next sections, the curvature term in Eq. (3) may have an important impact on the final 3D model created. This section presents its numerical discretization and thus the final numerical scheme for solving Eq. (3). The level set formulation of the mean curvature flow was suggested by Osher and Sethian (1998). The corresponding nonlinear partial differential equation has the following form:

$$u_t - |\nabla u| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) = 0 \tag{8}$$

where $u(x,t)$ = unknown level set function, $(x,t) \in \Omega \times (0,T)$; Ω = computational domain; and $(0,T)$ = time interval. The level set Eq. (8) is accompanied by the initial condition

$$u(x,0) = u_0(x), \quad x \in \Omega \tag{9}$$

Because of a possibly vanishing gradient in the case of arising flat areas, Eq. (8) is regularized by the so-called Evans-Spruck approach (Evans and Spruck 1991), leading to the equation

$$u_t - |\nabla u|_\varepsilon \nabla \cdot \left(\frac{\nabla u}{|\nabla u|_\varepsilon} \right) = 0 \tag{10}$$

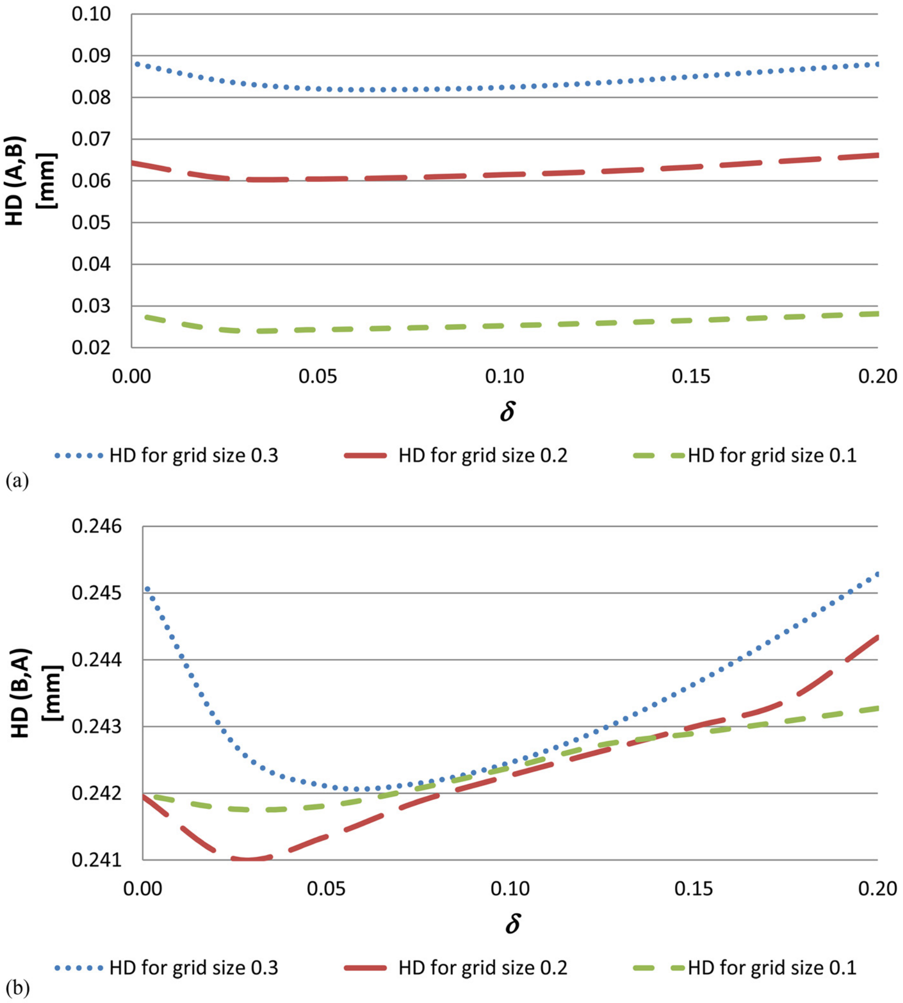

Fig. 6. Plots of (a) *HD(A,B)* and (b) *HD(B,A)* for the data set ch1 depending on the mesh density parameter h_D and the curvature parameter δ

Table 9. List of Objects, Their Size, Scanning Resolution, and Mesh Parameters

Object	Size of object (m)	Number of measured points	Resolution of scanning (mm)	Grid size h_D (mm)	Number of cells
Human vertebrae	0.09 × 0.08 × 0.05	1,133	3	0.5	160 × 150 × 140
Human vertebrae	0.09 × 0.08 × 0.05	2,424	2	0.4	200 × 200 × 140
Pillar in Gerulata	1.2 × 1.0 × 0.9	2,199	50	10	175 × 180 × 130
Well in Gerulata	2.8 × 2.8 × 1.1	3,704	50	10	285 × 285 × 110
Well in Gerulata	2.8 × 2.8 × 1.1	11,441	30	10	285 × 285 × 110
Part of generator	2.5 × 2.5 × 1.0	49,471	15	10	250 × 250 × 100
Part of statue	1.8 × 1.2 × 0.3	20,137	10	5	360 × 240 × 60
Church	16.5 × 5.5 × 13	421,067	70	50	360 × 260 × 280

where $|\nabla u|_\varepsilon = \sqrt{\varepsilon^2 + |\nabla u|^2}$; and ε = small regularization parameter.

The explicit finite-difference scheme for solving the regularized mean curvature flow level set Eq. (10), with time step (τ_S) and spatial grid size (h_D), is then given by

$$u^{n+1}_{i,j,k} = u^n_{i,j,k} + \tau_S \frac{\left(\begin{array}{l} \left(\varepsilon^2 + (u^n_{y_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2\right) u^n_{xx_{i,j,k}} + \left(\varepsilon^2 + (u^n_{x_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2\right) u^n_{yy_{i,j,k}} \\ + (\varepsilon^2 + u^n_{x_{i,j,k}} + u^n_{y_{i,j,k}})\, u^n_{zz_{i,j,k}} - 2u^n_{x_{i,j,k}} u^n_{y_{i,j,k}} u^n_{xy_{i,j,k}} \\ - 2u^n_{x_{i,j,k}} u^n_{z_{i,j,k}} u^n_{xz_{i,j,k}} - 2u^n_{y_{i,j,k}} u^n_{z_{i,j,k}} u^n_{yz_{i,j,k}} \end{array}\right)}{\varepsilon^2 + (u^n_{x_{i,j,k}})^2 + (u^n_{y_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2} \tag{11}$$

where the expressions $u^n_{x_{i,j,k}}$, $u^n_{y_{i,j,k}}$, $u^n_{z_{i,j,k}}$, $u^n_{xx_{i,j,k}}$, $u^n_{yy_{i,j,k}}$, $u^n_{zz_{i,j,k}}$, $u^n_{xy_{i,j,k}}$, $u^n_{xz_{i,j,k}}$, and $u^n_{yz_{i,j,k}}$ = central difference approximations of the first and second derivatives and are given by

$$u^n_{x_{i,j,k}} = \frac{u^n_{i+1,j,k} - u^n_{i-1,j,k}}{2h_D}, \quad u^n_{y_{i,j,k}} = \frac{u^n_{i,j+1,k} - u^n_{i,j-1,k}}{2h_D},$$

$$u^n_{z_{i,j,k}} = \frac{u^n_{i,j,k+1} - u^n_{i,j,k-1}}{2h_D},$$

$$u^n_{xx_{i,j,k}} = \frac{u^n_{i+1,j,k} - 2u^n_{i,j,k} + u^n_{i-1,j,k}}{h_D^2},$$

$$u^n_{yy_{i,j,k}} = \frac{u^n_{i,j+1,k} - 2u^n_{i,j,k} + u^n_{i,j-1,k}}{h_D^2}, \quad u^n_{zz_{i,j,k}} = \frac{u^n_{i,j,k+1} - 2u^n_{i,j,k} + u^n_{i,j,k-1}}{h_D^2}$$

$$u^n_{xy_{i,j,k}} = \frac{u^n_{i+1,j+1,k} + u^n_{i-1,j-1,k} - u^n_{i-1,j+1,k} - u^n_{i+1,j-1,k}}{4{h_D}^2},$$

$$u^n_{xz_{i,j,k}} = \frac{u^n_{i+1,j,k+1} + u^n_{i-1,j,k-1} - u^n_{i-1,j,k+1} - u^n_{i+1,j,k-1}}{4{h_D}^2},$$

$$u^n_{yz_{i,j,k}} = \frac{u^n_{i,j+1,k+1} + u^n_{i,j-1,k-1} - u^n_{i,j+1,k-1} - u^n_{i,j-1,k+1}}{4{h_D}^2}$$

The previous numerical scheme is explicit in time and thus not unconditionally stable; in practice, it is sufficient to use $\tau_S = h_D^2/4$ to keep stability of computations. If discretization Eqs. (7) and (11) are put together, the final numerical method for solving Eq. (3) containing both the advective and curvature terms is obtained

$$\begin{aligned} u^{n+1}_{i,j,k} = u^n_{i,j,k} - \frac{\tau_S}{h_D} & \left[\begin{array}{l} \max(-D^x_{i,j,k}d, 0)\,(u^n_{i,j,k} - u^n_{i-1,j,k}) + \min(-D^x_{i,j,k}d, 0)\,(u^n_{i+1,j,k} - u^n_{i,j,k}) \\ + \max(-D^y_{i,j,k}d, 0)\,(u^n_{i,j,k} - u^n_{i,j-1,k}) + \min(-D^y_{i,j,k}d, 0)\,(u^n_{i,j+1,k} - u^n_{i,j,k}) \\ + \max(-D^z_{i,j,k}d, 0)\,(u^n_{i,j,k} - u^n_{i,k,k-1}) + \min(-D^z_{i,j,k}d, 0)\,(u^n_{i,j,k+1} - u^n_{i,j,k}) \end{array}\right] \\ + \frac{\delta\ \tau_S}{\varepsilon^2 + (u^n_{x_{i,j,k}})^2 + (u^n_{y_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2} & \left(\begin{array}{l} \left(\varepsilon^2 + (u^n_{y_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2\right) u^n_{xx_{i,j,k}} + \left(\varepsilon^2 + (u^n_{x_{i,j,k}})^2 + (u^n_{z_{i,j,k}})^2\right) u^n_{yy_{i,j,k}} \\ + (\varepsilon^2 + u^n_{x_{i,j,k}} + u^n_{y_{i,j,k}})\, u^n_{zz_{i,j,k}} - 2u^n_{x_{i,j,k}} u^n_{y_{i,j,k}} u^n_{xy_{i,j,k}} \\ - 2u^n_{x_{i,j,k}} u^n_{z_{i,j,k}} u^n_{xz_{i,j,k}} - 2u^n_{y_{i,j,k}} u^n_{z_{i,j,k}} u^n_{yz_{i,j,k}} \end{array}\right) \end{aligned} \tag{12}$$

Quantitative Tests of the Algorithm

Choice of Grid Size

One important factor for model creation using LSMSR is a suitable choice of the grid size (h_D). Namely, it is important to choose an optimal size of h_D relative to a density of the input data points. If a very coarse grid is chosen, the final model will be rough and inaccurate. In contrast, if a very dense grid is chosen, the computation can take longer. To test the influence of the grid size, a smooth two-dimensional (2D) object with rounded tips was created (see Fig. 2). Three data sets with different densities of input points on the object boundary were considered (densities of 1, 2, and 3 mm). They are referred to as ch1, ch2, and ch3. Then, various grid sizes were chosen varying from 0.05 to 0.5 mm with the step 0.05. First, the authors set $\delta = 0$, which means any curvature influence was excluded. In the top row of Fig. 2, testing data sets ch1, ch2, and ch3 are plotted, and below (in columns *a*, *b*, and *c*), the models generated by LSMSR are plotted with h_D = 0.05, 0.1, 0.2, 0.3, and 0.5 mm.

By a visual comparison, one can see that models created by using the grid size $h_D = 0.5$ are too rough and unsmooth in all three cases. The best results were obtained by using $h_D = 0.1$ and

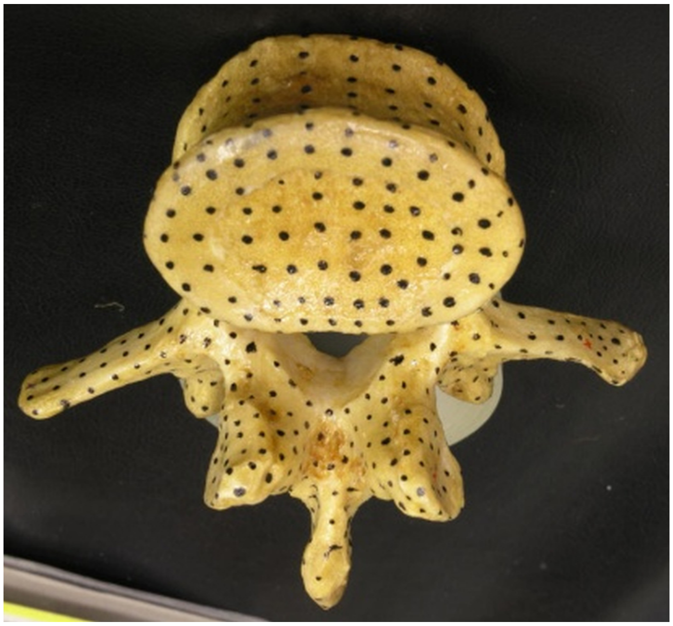

Fig. 7. Picture of the human vertebrae

$h_D = 0.05$. However, for $h_D = 0.05$, the overall computations took several times longer than for $h_D = 0.1$. In contrast, both results were visually comparable and smooth. One can also see that, for the coarser data sets ch2 and ch3, the coarser grid sizes $h_D = 0.2$ and $h_D = 0.3$ give visually good results, almost comparable regarding smoothness with $h_D = 0.1$. Such visual comparisons clearly indicate that the choice of grid size h_D has a relation with density of input points and influences the quality of the final model created by LSMSR. To get a quantitative insight into this fact, the authors computed the directed Hausdorff distance between points of input data sets ch1, ch2, and ch3 and the points generated as an intersection of the level set (isoline in this 2D testing example), representing the final model, with the grid lines of the mesh.

Let $A = \{a_1, a_2, \ldots, a_p\}$ be a set of input points and $B = \{b_1, b_2, \ldots, b_q\}$ be a set of points of the level set representing the final model

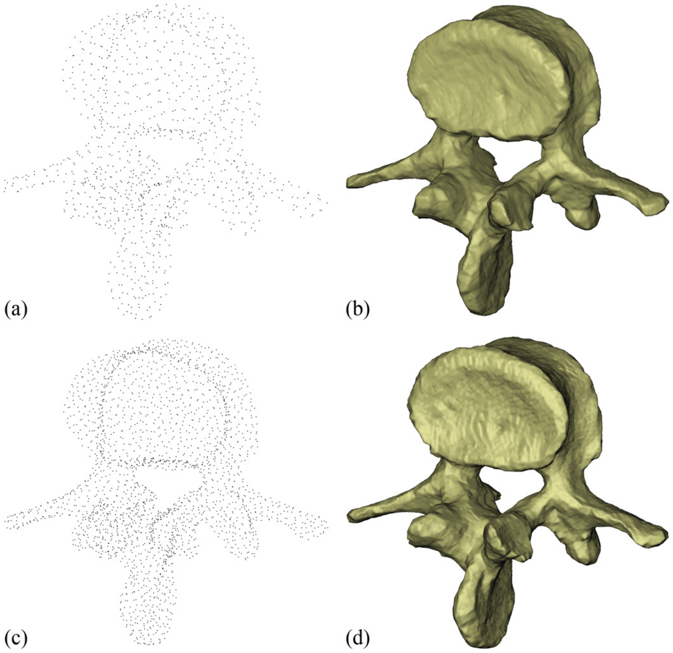

Fig. 8. Human vertebrae: (a) point cloud with scanning resolution of 3 mm; (b) model created by LSMSR; (c) point cloud with scanning resolution of 2 mm; (d) model created by LSMSR

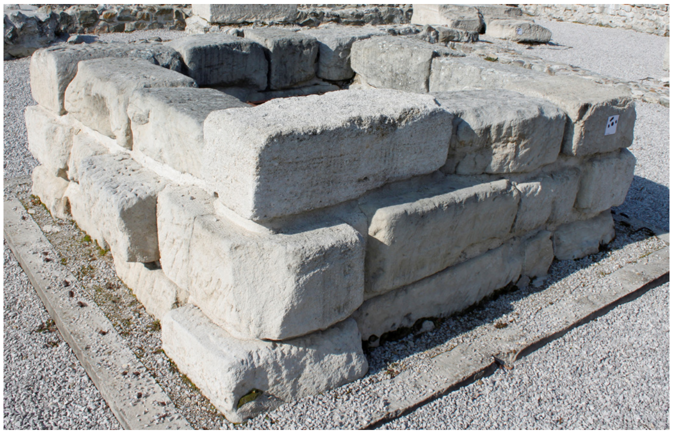

Fig. 10. Picture of the well in Gerulata

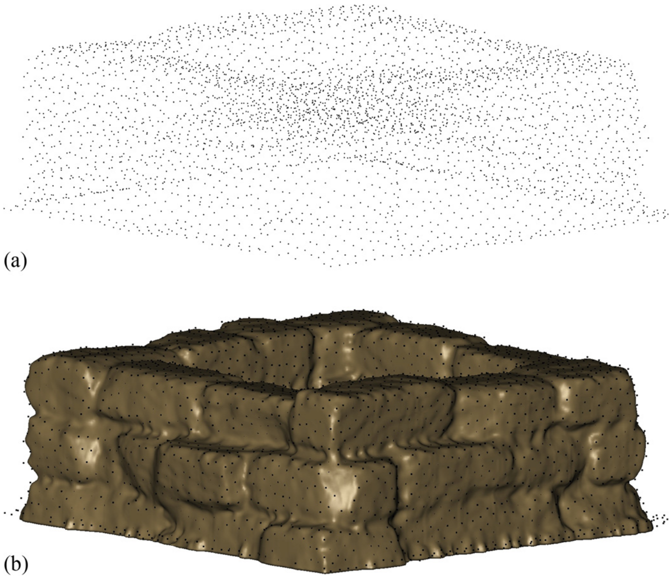

Fig. 11. Well in Gerulata: (a) point cloud reduced to point density 5 cm; (b) model created by LSMSR

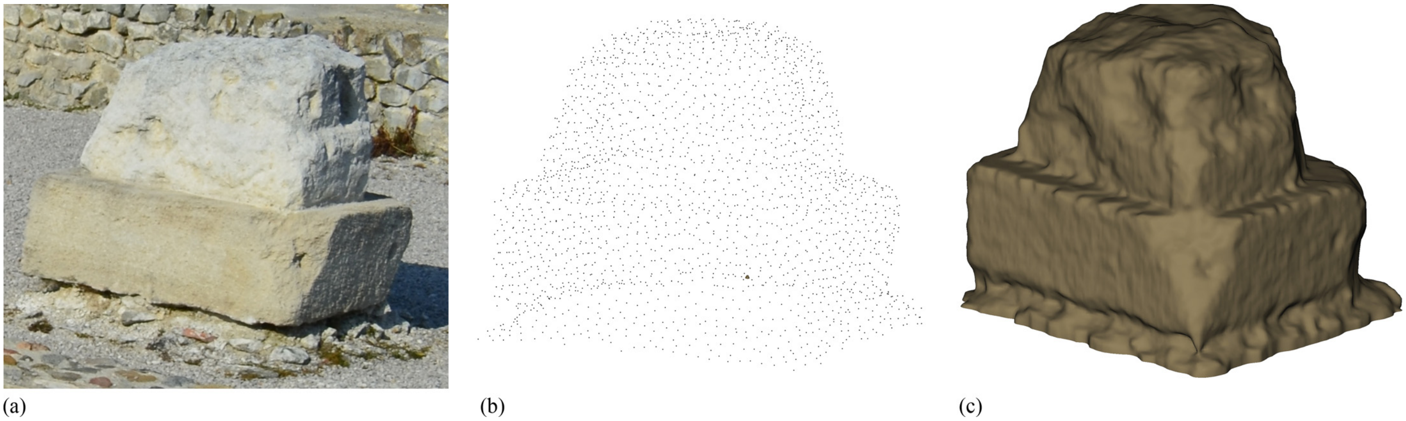

Fig. 9. Pillar in Gerulata: (a) picture; (b) point cloud with scanning resolution of 3 cm; (c) model created by LSMSR

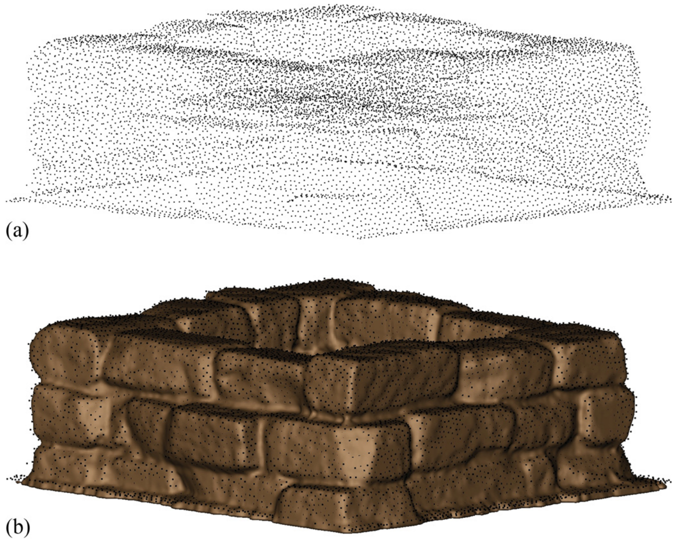

Fig. 12. Well in Gerulata: (a) original point cloud; (b) model created by LSMSR

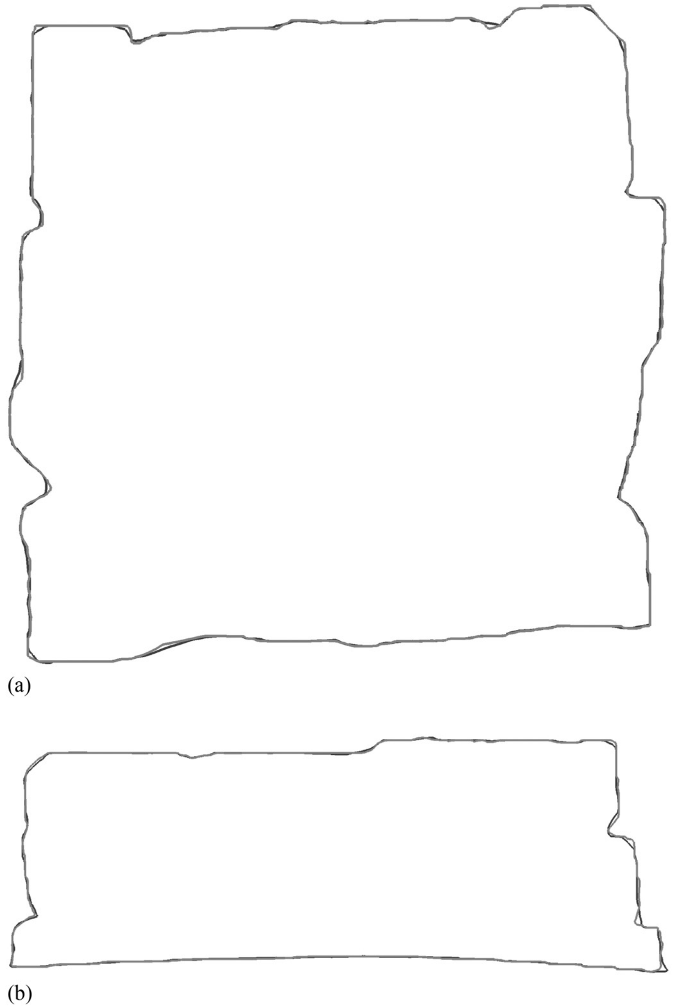

Fig. 13. 2D slices of the model created by LSMSR from the original point cloud (gray) and the reduced point cloud (black): (a) top view; (b) side view

generated as described earlier. The directed averaged Hausdorff distances are then defined as follows:

$$HD(A,B) = \frac{1}{p}\sum_{i=1}^{p}\min_{b\in B}\|a_i - b\|$$

$$HD(B,A) = \frac{1}{q}\sum_{i=1}^{q}\min_{a\in A}\|b_i - a\|$$

where ‖…‖ = Euclidean distance. For every model ch1, ch2, and ch3, *HD(A,B)* and *HD(B,A)* were computed. In the first case, for every input point of data set, the authors looked for the closest interpolated point of the isoline representing the final model, and then the average of these closest distances was obtained. In the second case, for every interpolated point, the authors looked for the nearest point in the input data set and averaged these distances. As expected, in the first case, the *HD(A,B)* decreases with increasing mesh density (see Table 1), because there are more interpolated points in the vicinity of any data point when refining the mesh. This table also shows correct behavior of LSMSR; by increasing the mesh density, the final model is always closer to the data points. A further important observation is presented in Table 2, which shows quantitatively the fact that there exists an optimal h_D for every data set ch1, ch2, and ch3. It clearly shows that the optimal isoline (with the smallest Hausdorff distance to the set of input points) is obtained by using h_D approximately equal to 1/5 to 1/10 of the input data set density. A further mesh refinement does not bring any improvement in the Hausdorff distance *HD(B,A)*.

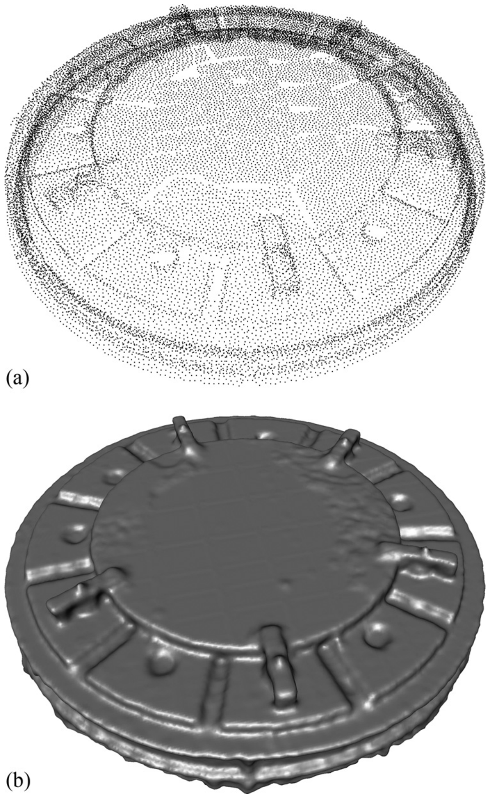

Fig. 14. Part of the turbine generator in Gabčíkovo: (a) point cloud; (b) model created by LSMSR

Influence of Curvature

In the case of the curvature term included in the model [Eq. (3)], the shape of reconstructed surface is influenced by the choice of parameter (δ). To test the influence of this parameter, a similar strategy as described earlier and the same input data sets ch1, ch2, and ch3 were used. Again, various mesh densities (h_D = 0.1, 0.2, 0.3, and 0.5 mm) were used, and the authors set $\delta = 0.0$, 0.25, 0.5, 0.75, and 1.0. Fig. 3 explores obtained results visually, considering the input data set ch1. Clearly, a too-high value of δ may lead to undesired smoothing of the final model, especially when using coarser meshes ($h_D = 0.3$ and $h_D = 0.5$). In concave parts of the desired object, the final curve moved too much inside the object, and in a convex part, it does not move sufficiently close to the given data points. In contrast, when using these coarse meshes, one can see that with $\delta = 0.0$, the final shape is too rough, and some smoothing would be desirable. Thanks to this visual inspection, the authors decided to study the influence of the curvature parameter (δ) in more detail, using a smaller range of 0.0–0.2 and a smaller step of 0.025. The results are reported in Tables 3–8 and Figs. 4–6.

The results show that by using a nonzero δ, a smaller *HD(A,B)* can be obtained, as with $\delta = 0.0$, and $\delta = 0.05$ seems to be an optimal choice in all cases [see Tables 3, 5, and 7 and Figs. 4–6(a)].

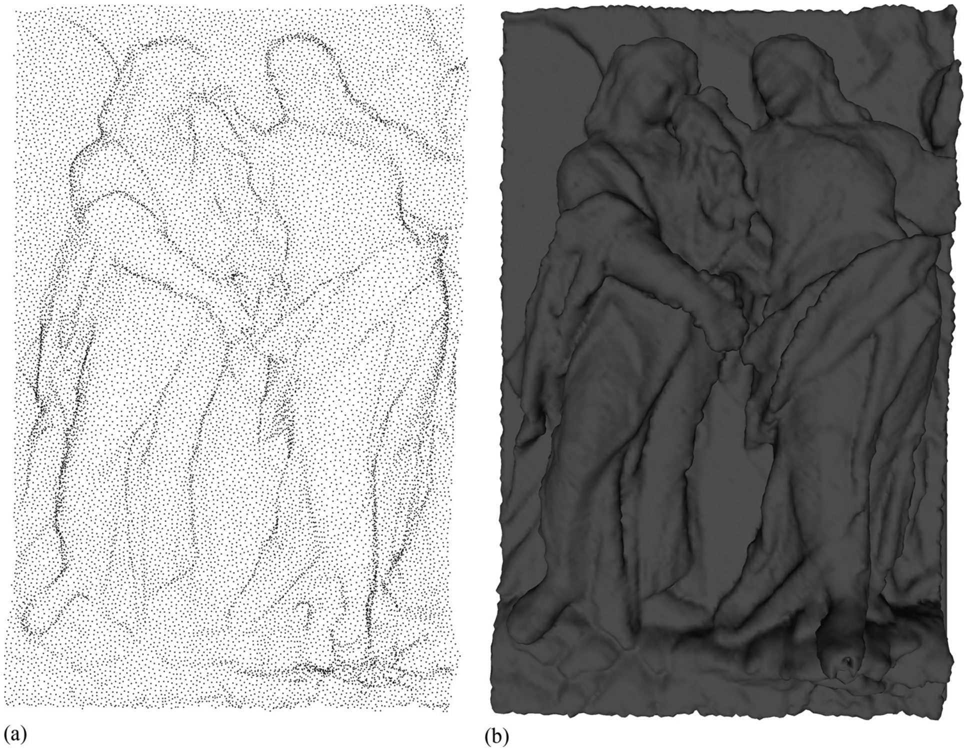

Fig. 15. Part of the statue: (a) point cloud; (b) model created by LSMSR

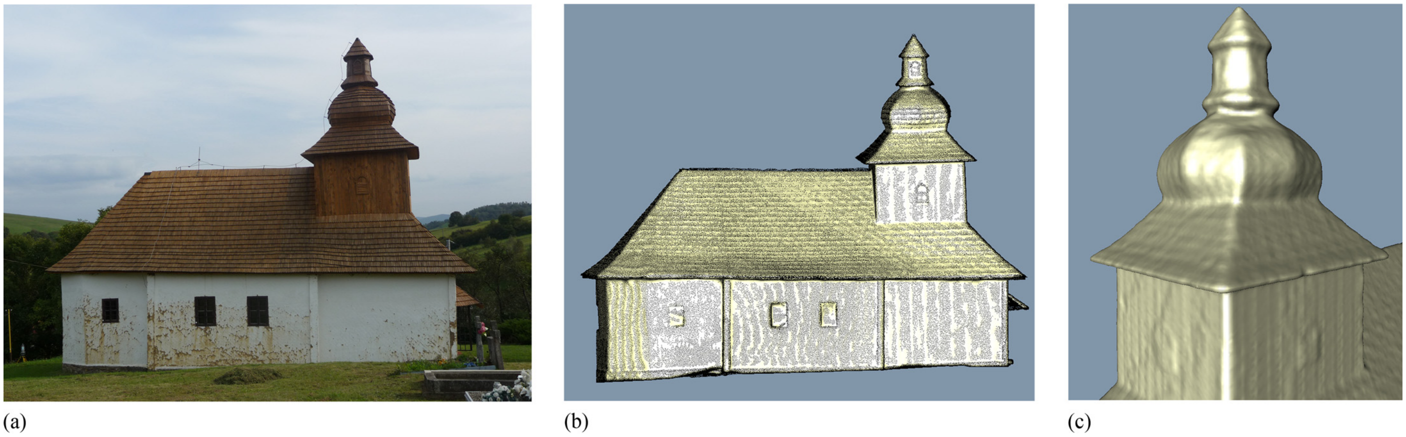

Fig. 16. Wooden Greek Catholic Church of the Nativity of St. John the Baptist in Kalná Roztoka: (a) picture (image by Jana Haličková); (b) model created by LSMSR; (c) detail of the model

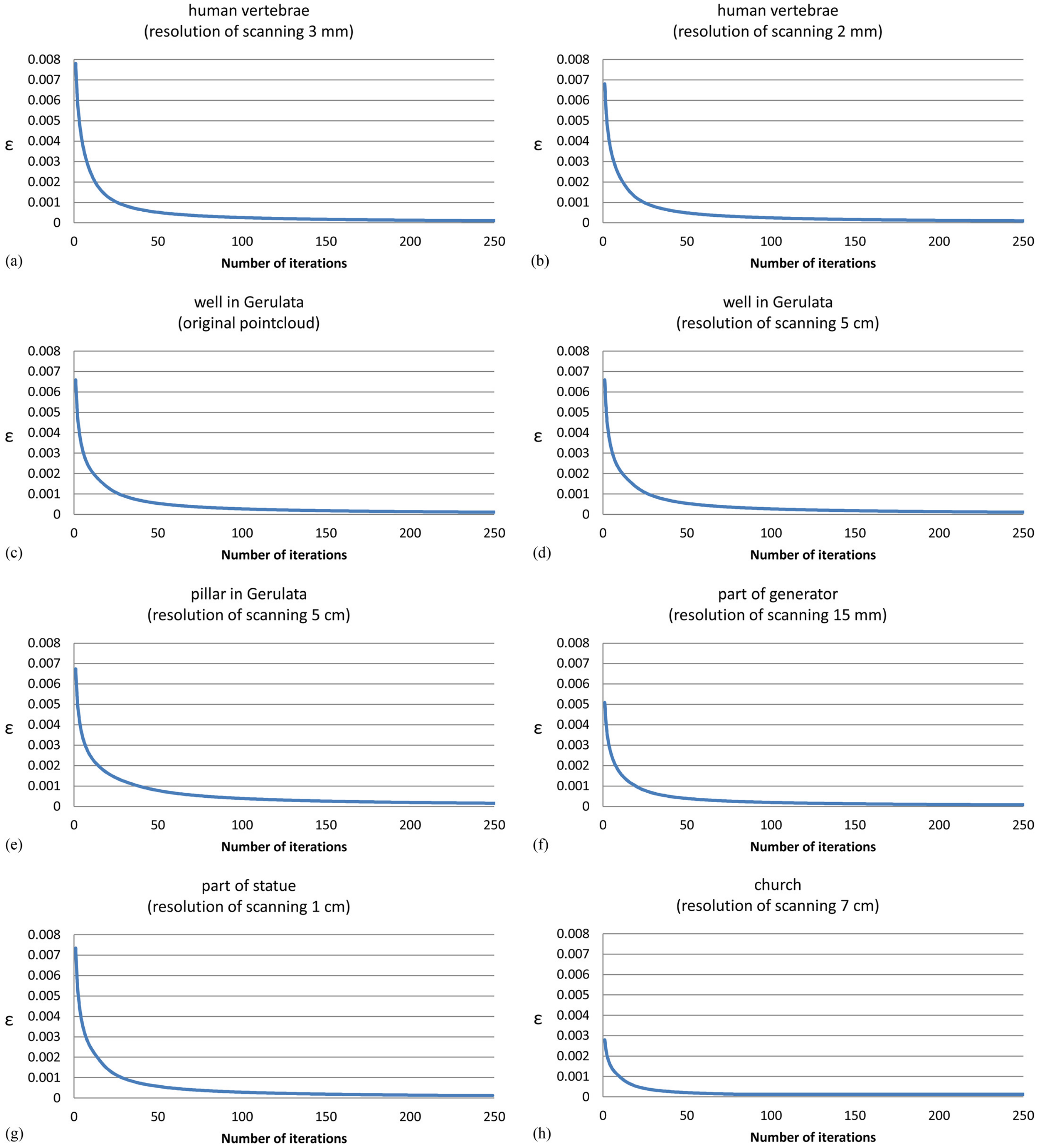

Fig. 17. Plots of ε depending on the number of time steps

Tables 4, 6, and 8 and Figs. 4–6(b) show that, for the choice of δ in the range of 0.025–0.075, the values of *HD(B,A)* are almost nonaltered when h_D is equal to 1/10 of the density of the input data set. In contrast, in the case of coarser grids (i.e., with a higher h_D), the values of *HD(B,A)* are significantly lower for δ from that range of values. The results also show that, by using nonzero δ, better results can be obtained with coarser meshes than with $\delta = 0.0$ on finer meshes. Usage of the coarser mesh saves the memory and computational time requirements of the LSMSR algorithm, and thus it is reasonable to use δ around 0.05 when applying the method in practical examples.

Practical Examples of 3D Model Creation Using LSMSR

This section presents some numerical examples of model creation that illustrate the quality of the surface reconstruction. Data points of human vertebrae were obtained by optical scanning using a COMET system (Steinbichler , Neubeuern, Germany). The data points of objects in the archaeological site Gerulata, the turbine generator in the Gabčíkovo hydroelectric power plant, and the wooden Greek Catholic Church of the Nativity of St. John the Baptist in Kalná Roztoka were obtained by laser scanning using Leica ScanStation (Leica Geosystems AG, Heerbrugg, Switzerland), and part of the statue was obtained by photogrammetry using a Nikon D200 camera (Nikon, Tokyo). Information about objects, such as the size, the resolution of scanning, and the chosen parameters for model creation, is presented in Table 9.

Fig. 7 shows the picture of human vertebrae. Figs. 8(a and c) show the data points in two resolutions, and Figs. 8(b and d) show models created by LSMSR. In these two cases, the data sets are quite uniform. The model created from the higher-resolution data set is very detailed, and the difference between these two models is very small.

Fig. 9 shows the reconstruction of the pillar in Gerulata from the low-resolution data set. In this case, the algorithm easily handles this type of data points, and the model is smooth and captures the topology of the object surface very well.

Next, the reconstruction of the well in Gerulata is shown, plotted in Fig. 10, by using two different resolutions of the data. Figs. 11(a and b) show the reduced data points with a resolution of 5 cm and the model created from this data set. Figs. 12(a and b) show the original data points with a resolution higher than 3 cm and the model created from this data set. Fig. 13 shows selected 2D slices of both created models to see quantitatively a difference in reconstructions. Only small differences are observed; the model created from the original data set is more detailed and smoother.

Figs. 14–16 show other applications of model creation using LSMSR. Fig. 14 shows the possibility of using LSMSR for model creation from a quite nonuniform data set in an industrial application, and Fig. 15 shows nonuniform data points of part of a statue and the model created from this data set. Fig. 16 shows a picture of the wooden Greek Catholic Church of the Nativity of St. John the Baptist built in 1750, the created model together with the scanned point cloud and details of the reconstructed surface.

In all presented numerical experiments, finding an initial guess was extremely fast, taking only a few seconds. After finding the initial guess, the numerical model was first used without the curvature term (100–150 time steps), and then the model was used with the curvature term with a small δ to finish the surface reconstruction (5–10 time steps). Because the values of evolving level set function are within the interval [0,1], the final reconstructed shape is obtained as the isosurface 0.5 (i.e., middle value) of the final shock-like profile of the level set function.

To get a right choice of the number of time steps for the model creation, a change in the level set function (u) after each time step was tested. The mean of squared differences of two subsequent time steps was computed

$$\varepsilon = \frac{1}{N} \sum_{i,j,k} (u^{n+1}_{i,j,k} - u^{n}_{i,j,k})^2$$

where N = number of all grid points. The evolution of ε with respect to time is reported in Fig. 17. In all presented experiments, after approximately 100 time steps, ε drops below 1/10 to 1/100 of the grid size, which indicates a possibility to stop the evolution.

Table 10. Values of Hausdorff Distance for All Models Created by LSMSR

Object	Resolution of scanning (mm)	Grid size h_D (mm)	Number of time steps	HD (mm)
Human vertebrae	3	0.5	100	0.22
Human vertebrae	2	0.4	100	0.17
Pillar in Gerulata	50	10	150	5.9
Well in Gerulata	50	10	150	9.0
Well in Gerulata	30	10	150	7.9
Part of generator	15	10	150	6.3
Part of statue	10	5	150	2.1
Church	70	50	150	15

Note: HD = Hausdorff distance.

To get a quantitative insight into the quality of the reconstruction of real objects, the authors computed the directed Hausdorff distance between points of the point cloud of all objects and points generated as the intersection of the 0.5 isosurface, representing the final model, with the grid lines of the mesh. The results are reported in Table 10.

Conclusions

This paper presents a new LSMSR and its application to 3D model creation from data sets obtained by laser scanning and photogrammetry. The mathematical models and numerical methods used in LSMSR are described. The authors also studied the influence of method parameters on the final reconstructed shape and presented real examples of LSMSR application in surveying. In all cases, the obtained results show good behavior of the new method with respect to efficiency and quality of reconstruction.

The final reconstructed shape is obtained as an isosurface of the equilibrium state (shock profile, similar to the subjective surface method) of the evolving level set function. The quality of reconstruction was tested by using the directed mean Hausdorff distance between such isosurface and the given data point cloud. Such tests show that it is optimal to choose a computational mesh density of approximately 1/5 to 1/10 of the data point density, and that usage of a small curvature parameter helps in getting smoothness of the shape without its distortion. By using curvature regularization, good quality results can be obtained on coarser computational grids, thus saving the memory, which can be critical in large-scale applications.

Acknowledgments

This work was supported by grant APVV-0072-11.

References

Bourgine, P., Frolkovic, P., Mikula, K., Peyrieras, N., and Remesikova, M. (2009). “Extraction of the intercellular skeleton from 2D microscope images of early embryogenesis.” *Proc., 2nd Int. Conf. on Scale Space and Variational Methods in Computer Vision*, Springer-Verlag, Berlin, 38–49.

Corsaro, S., Mikula, K., Sarti, A., and Sgallari, F. (2006). "Semi-implicit co-volume method in 3D image segmentation." *SIAM J. Sci. Comput.*, 28(6), 2248–2265.

Dorninger, P., and Pfeifer, N. (2008). "A comprehensive automated 3D approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds." *Sensors*, 8(11), 7323–7343.

El Meouche, R., Rezoug, M., Hijazi, I., and Maes, D. (2013). "Automatic reconstruction of 3D building models from terrestrial laser scanner data." *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, II-4(W1), 7–12.

Evans, L. C., and Spruck, J. (1991). "Motion of level sets by mean curvature." *Differ. Geom.*, 33(1991), 635–681.

Mikula, K., and Sarti, A. (2007). "Parallel co-volume subjective surface method for 3D medical image segmentation." *Deformable models: Theory and biomaterial applications*, J. S. Suri, and A. Farag, eds., Vol. II, Springer Science+Business Media, Berlin, , 123–160.

Osher, S., and Fedkiw, R. (2002). *Level set methods and dynamic implicit surfaces*, Springer-Verlag, New York.

Osher, S., and Paragios, N. (2003). *Geometric level set methods in imaging, vision and graphics."* Springer-Verlag, New York.

Osher, S., and Sethian, J. A. (1988). "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations." *J. Comput. Phys.*, 79(1), 12–49.

Rouy, E., and Tourin, A. (1992). "Viscosity solutions approach to shape-from-shading." *SIAM J. Numer. Anal.*, 29(3), 867–884.

Sarti, A., Malladi, R., and Sethian, J. A. (2000). "Subjective surfaces: A method for completing missing boundaries." *PNAS, 97(12)*, 6258–6263.

Sethian, J. A. (1999). *Level set methods and fast marching methods: Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, Cambridge University Press, Cambridge, U.K.

Zhao, H. K., Osher, S., and Fedkiw, R. (2001). "Fast surface reconstruction using the level set method." *Proc., IEEE Workshop on Variational and Level Set Methods in Computer Vision* (VLSM 2001), New York.

Zhao, H. K., Osher, S., Merriman, B., and Kang, M. (2000). "Implicit and non-parametric shape reconstruction from unorganized data using a variational level set method." *Comput. Vision Image Understanding*, 80(3), 295–319.