Using Anti-Aliased Signed Distance Fields for Generating Surgical Guides and Plates from CT Images

Fredrik Nysjö, Pontus Olsson, Filip Malmberg, Ingrid B. Carlbom, and Ingela Nyström
Centre for Image Analysis, Dept. of Information Technology,
Uppsala University, Sweden
{fredrik.nysjo, pontus.olsson, filip.malmberg, ingrid.carlbom, ingela.nystrom}@it.uu.se

ABSTRACT

We present a method for generating shell-like objects such as surgical guides and plates from segmented computed tomography (CT) images, using signed distance fields and constructive solid geometry (CSG). We develop a user-friendly modeling tool which allows a user to quickly design such models with the help of stereo graphics, six degrees-of-freedom input, and haptic feedback, in our existing software for virtual cranio-maxiollofacial surgery planning, HASP. To improve the accuracy and precision of the modeling, we use an anti-aliased distance transform to compute signed distance field values from fuzzy coverage representations of the bone. The models can be generated within a few minutes, with only a few interaction steps, and are 3D printable. The tool has potential to be used by the surgeons themselves, as an alternative to traditional surgery planning services.

Keywords

Implicit Modeling, Distance Fields, CT, Shells, Virtual Surgery Planning

1 INTRODUCTION

Virtual surgery planning and pre-operative design and fabrication of plastic surgical guides and titanium plates for bone fixation have proven valuable for improving outcome and reducing cost in reconstructive surgery such as cranio-maxillofacial (CMF) surgery [25]. However, existing virtual surgery planning tools, such as Materialise's Mimics [14], have complex user interfaces (UIs) limited to two-dimensional (2D) interaction with three-dimensional (3D) data, such as computed tomography (CT) images, which can be non-intuitive for clinicians to use and therefore often require the help of a technician or an engineer. This includes the task of designing of patient-specific models of surgical guides and plates, something which today is often outsourced to external companies with lead times of several days or in some cases weeks.

The Haptics-Assisted Surgery Planning (HASP) system [18][19] was developed within our research group with the aim of shortening the preoperative planning time from days to hours, by providing a user-friendly software and UI that can be used by the surgeons themselves. HASP supports stereo graphics, six degrees-of-freedom (DOF) input, and haptic feedback in a software that incorporates bone, vessels, and soft tissue planning for CMF defect reconstruction. Surgeons who have tested the software have found it to be an efficient tool for planning so called fibula osteocutaneous free flap reconstructions, in which a bone graft from the fibula (calf bone) is transplanted to the defect, for example, a resected part of the mandible (lower jaw) [19].

Previous versions of the HASP software did not support the design of patient-specific plates for bone fixation, or cutting guides with slots or flanges that help the surgeon to perform precise osteotomies or resections (cuts) according to a surgical plan. In this paper, we present the tools we developed to enable surgeons to quickly design models for such parts within a few minutes, and also present a method for generating the models from segmented CT images.

1.1 Contribution

The main contributions of this paper are:

- A fast and efficient method for generating 3D printable models of surgical guides and plates from segmented CT images.
- A method for improving the accuracy and precision of the modeling by using fuzzy coverage representations in combination with an anti-aliased distance transform.

2 RELATED WORK

There are a few examples in the literature of systems or methods for helping surgeons designing surgical guides

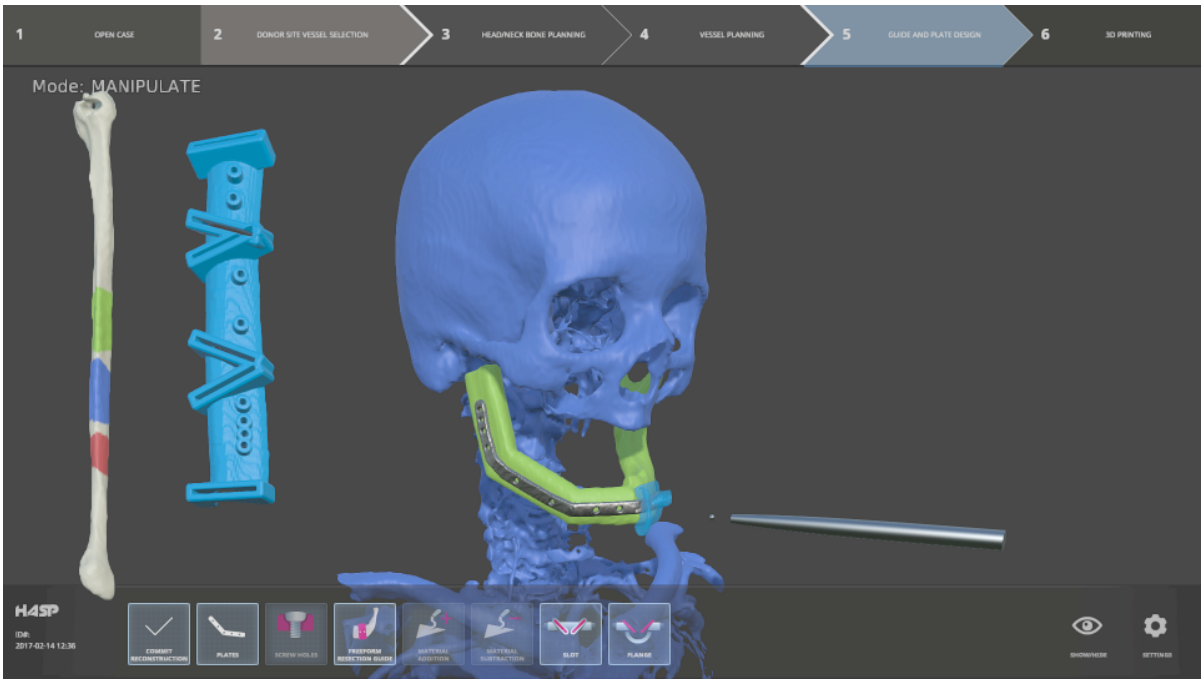


Figure 1: The HASP user interface. A six DOF haptic device controls the virtual stylus (right) used for interaction. The screenshot shows a mandible (in green) being reconstructed from the original mandible combined with a transplant from the fibula (left), and surgical guides (in blue) and plates (in metal) generated in the software.

and plates. Kovler et al. [12] describe a system combining stereo graphics and haptic feedback with a sketch-based method for plate design. However, their system is aimed at trauma surgery and does not provide generation of surgical guides. Fornaro et al. [7] present another approach to plate design, in which a virtual plate model is deformed for planning pre-bending of physical plates. Voss et al. [24] describe a system for fracture reduction that allows design of simple surgical guides.

The previous examples use polygonal, or mesh-based, representations for the modeling. Volumetric, or voxel-based, representations can be more suitable for organic shapes such as the human anatomy, and make it easier to guarantee a 3D printable result when computing for example surface offsets. Geomagic Freeform [9] is an example of a commercial voxel-based sculpting software that allows creating patient-specific parts such as surgical guides and implants. It provides a traditional 2D UI, combined with six DOF input and haptic feedback for more intuitive 3D interaction. However, the software is too complex for most clinicians to use, as it is not aimed specifically at surgery planning.

For an overview of state-of-the-art in virtual surgery planning and implant design, see Ritacco et al. [23].

3 METHODS

Our approach is to extract a shell (Figure 2f) around the bone that serves as template for the part the surgeon wishes to design, i.e., a surgical guide or plate, from a grayscale CT (Figure 2a) and a binary segmentation (Figure 2b) of the bone, and generate a constructive solid geometry (CSG) tree that includes the shell and other components of the part. The CT and segmentation data are loaded as volume images. The shapes for the shell and other components in the CSG tree are represented as signed distance fields (SDFs) computed from the image data (Figures 2d–2e). Other inputs for generating the CSG tree include osteotomy and resection planes from the planning and user-generated inputs such as control points for defining geometry. By evaluating the CSG tree, we obtain an image from which a 3D printable triangle mesh can be extracted and, finally, exported to stereolithography (STL) mesh format.

3.1 Signed Distance Fields

A signed distance field (SDF) maps a point to a positive or negative distance scalar value, depending on whether the point lies inside or outside an object in the image. While SDFs have many applications [11], we are mainly interested in their use for implicit solid modeling. An early example of such use is found in Payne et al. [21], who describe operations such as Boolean set operations. A more recent example is the framework presented by Museth et al. [16].

3.2 Anti-Aliased Distance Transform

A distance transform (DT) is an efficient way of computing SDFs from binary images. Among approximate (non-Euclidean) 3D DTs, the Chamfer DT with integer

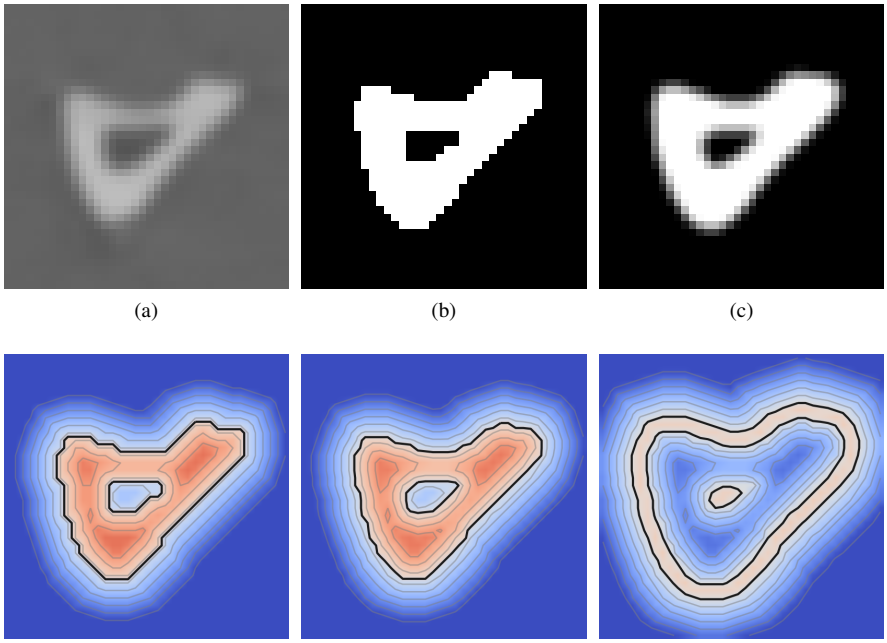


Figure 2: 2D slices of 3D image data for a fibula: (a) grayscale CT; (b) binary segmentation; (c) fuzzy coverage representation; (d) SDF from binary DT; (e) SDF from AA DT; (f) shell SDF computed from (e). In (d–f), the black iso-contour represents the zero-level, whereas the color indicate positive (red) or negative (blue) distance.

weights $\langle 3,4,5\rangle$ is fast to compute and has a bounded error of 11.8% to the Euclidean distance [2]. Efficient algorithms for computing the Euclidean DT in 3D include the vector propagation algorithm by Danielsson [5] and the more recent method by Felzenszwalb and Huttenlocher [6], both which can be parallelized for faster computation.

We implemented the Chamfer $\langle 3,4,5\rangle$ DT algorithm, since it is easy to implement, and because accurate distances are only needed close (within a few millimeters) to the bone template surface; we also aim to keep the memory requirements for computing and storing the SDFs low. We adopt the convention that voxels inside objects have positive values and voxels outside objects have negative values, and compute the SDF in two passes, one for the internal distances and one for the external distances. The computed distance values are stored as 8-bit signed integers, such that integer distances are constrained (clamped) to the range $[-128,127]$.

Anti-aliased (AA) DTs can produce more accurate distance values compared to binary DTs, in particular close to object surfaces, by using sub-pixel or sub-voxel information in grayscale image data. Other benefits are a smoother iso-surface for visualization and triangle mesh extraction, and a gradient magnitude closer to 1 near the zero-level isosurface (Figure 2e). Gustavson and Strand [10] propose an extension of Danielsson's [5] vector propagation algorithm, in which they incorporate pixel coverage information when computing the DT; they suggest a fast linear approximation d_f for mapping coverage values to sub-pixel (or sub-voxel) distances,

$$d_f = 0.5 - a, \tag{1}$$

where a is a fuzzy coverage value in the range $[0,1]$.

To retain the performance of computing the Chamfer DT, we extend our Chamfer DT implementation to incorporate voxel coverage information, instead of implementing the AA Euclidean DT in [10]. In each pass (interior and exterior), we use Equation 1 to initialise foreground voxels that otherwise would be assigned the value zero with sub-voxel distances computed from coverage, in order to compute an AA DT.

3.3 Binary-to-Coverage Conversion

CT images exhibit fuzzy tissue boundaries due to the partial volume effect (PVE), and also from filtering for removing noise. Segmented CT data, in contrast, often provide binary label masks. When the binary segmentation is computed by thresholding and a subsequent labeling step, for example using the method by Nysjö et al [17], so that foreground voxels in label masks correspond to foreground voxels in the original thresholded image, we can convert the binary label masks to fuzzy coverage representations suitable as input to an AA DT.

To perform binary-to-coverage conversion, we extract a 2-voxel thick boundary around each labeled object, containing the interior and exterior boundary voxels computed for 27-connectivity. At each boundary voxel,

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

Figure 3: Fibula cutting guides: (a–b) slot type (front and back); (c–d) flange type (front and back).

we supersample the original grayscale CT image by trilinear interpolation at 4^3 (64) sub-voxel positions in a uniform grid, and compute coverage as $a = \frac{1}{N}\sum_{i=0}^{N-1} x_i$, where N is the number of samples and $x_0,\ldots,x_{N-1}$ are thresholded sample values. The resulting fuzzy coverage representation (Figure 2c) allows sub-voxel distances to be estimated at a resolution of $\frac{1}{4}$ voxel, using Equation 1.

3.4 Shell Generation

In order to generate a surgical guide or plate that fits the anatomy of the patient, we should compute a shell at an accurate and precise offset from the bone template. We generate the shell by applying the function

$$d_{shell}(x) = max(d_{inner} - x, x - d_{outer}), \qquad (2)$$

where d_{inner} and d_{outer} define the distance offsets from the original iso-surface to the inner and outer iso-surfaces of the shell, respectively, to the SDF of the bone template (Figure 2d or 2e), resulting in a new SDF for the shell (Figure 2f).

3.5 Constructive Solid Geometry

Constructive solid geometry (CSG) provides a compact representation for Boolean set operations (unions, intersections, and differences) on solid shapes, by storing shapes and operations as nodes in a binary tree. It can be used for implicit solid modeling with SDFs, by using Boolean set operators for SDFs [11] and images for storing shapes and intermediate results.

For each guide or plate model, we generate a CSG tree containing the shell and other components needed for the part, as described in the next three sections. SDFs for other components are either computed from voxelised triangle meshes or sampled from distance functions of basic primitives [22].

As a final step, when the CSG result image should only contain a single connected component, we use connected component analysis to remove smaller components, such as unwanted material generated where the bone template is hollow.

3.5.1 *Fibula Cutting Guides*

Fibula cutting guides are automatically created when the surgeon presses a button. The cut planes for the fibula osteotomies are used to cut the shell SDF of the bone template, the number of cut planes used determined by which type of guide (slots or flanges) the surgeon wishes to create (Figure 3). When generating the CSG tree, we insert components for slots and flanges, bridges for connecting the guide segments, and drill guides for the plate screw holes. We also compute the difference with the bone template SDF after adding the d_{inner} shell offset to the SDF, to remove unwanted material inside the guide. Evaluating a CSG tree takes a few seconds, which allows the surgeon to try different designs and adjust the orientation of the guide.

3.5.2 *Plates*

To create plates, the surgeon uses the stylus to place control points on the surface of the bone template, where the plate should be in contact with the bone (Figure 4a), and also uses the stylus to place markers for screws (Figure 4b). The plate is updated at interactive speed, and existing control points and screw markers may be moved or deleted with the stylus. The SDF of the bone template is also used as an aid to snap control points to the surface and automatically orient screw markers in the SDF gradient direction. When generating the CSG tree, we insert spheres interpolating the control points, and holes at the screw markers. We also compute the intersection with the shell SDF from the bone template, to generate the final plate (Figure 4c).

3.5.3 *Mandibular Resection Guides*

A 3D brush (Figure 5a) is used to define the geometry of the mandibular resection guides (Figure 5b). The surgeon controls the brush with the stylus, and switches between additive and subtractive brushes. When generating the CSG tree, we insert spheres for brush strokes. We also compute the intersection with the shell SDF from the bone template, and insert drill guides for the plate screw holes (Figure 5c).

(a) (b) (c)

Figure 4: Plate design: (a) plate geometry (in blue) interpolated along control points the user places on the bone template (in green) using the stylus; (b) markers for screws; (c) final plate with screw holes inserted.

(a) (b) (c)

Figure 5: Mandibular resection guide design: (a) 3D brush tool; (b) painted guide geometry; (c) final resection guide with flanges and drill guides inserted.

3.6 Visualization and Haptic Rendering

We use Marching cubes [13] (MC) to visualize the iso-surfaces of the bone templates and the models from the generated CSG trees. For the CSG trees, we utilize a sparse block data structure to keep track of blocks that contain modified data and update the iso-surface only in those blocks during modeling. The MC iso-surfaces are also exported to STL mesh format for 3D printing.

For the haptic rendering, we convert bone fragments into so called Voxmap Pointshell [15] representations. This enables haptic interaction between bone surfaces and the stylus, as well as haptic interaction between bone fragments. Further details about the haptic rendering are provided in [18].

3.7 Implementation Details

We implement the tools for surgical guide and plate generation in the HASP software, using C++ and our own OpenGL 3.2 (Core profile) based rendering framework. OpenMP and single instruction/multiple data (SIMD) intrinsics are used to speed up the image processing and the CSG operations. Our MC implementation is based on the code available at [3].

4 EXPERIMENTS AND RESULTS

To evaluate our method, we performed simulated surgery planning on three cases. Each case included CT or CT angio (CTA) images, in DICOM format, of the head and neck region and of the leg from which the bone transplant was to be harvested. The test datasets are summarized in Table 1. The dataset for Case 1 consisted of the MANIX and OBELIX images from the public OsiriX DICOM repository [20]; the OBELIX image was cropped to include only the left leg. The datasets for Cases 2 and 3 consisted of images from actual patients that were about to undergo reconstructive surgery at the time of the imaging.

We used the BoneSplit [17] software to convert the DICOM image stacks to volume images in VTK format and to segment the mandibles from the head-neck CT images and the fibulas from the leg CT images. For bone thresholding, we used threshold values in the range $[146, 261]$ Hounsfield units (HU), corresponding to conservative values for bone. We obtained cut planes for fibula osteotomies and mandible resections from the planning software, and also computed fuzzy coverage representations of the reconstructed mandibles. Images with anisotropic voxels were resampled to isotropic voxels before we computed the SDFs. After consulta-

CT	Size (voxels)	Voxel spacing (mm)
Head 1	$512 \times 512 \times 460$	$0.49 \times 0.49 \times 0.70$
Leg 1	$225 \times 241 \times 443$	$0.74 \times 0.74 \times 1.00$
Head 2	$512 \times 512 \times 159$	$0.30 \times 0.30 \times 1.00$
Leg 2	$137 \times 141 \times 135$	$0.88 \times 0.88 \times 3.00$
Head 3	$512 \times 512 \times 409$	$0.40 \times 0.40 \times 0.60$
Leg 3	$157 \times 149 \times 613$	$0.63 \times 0.63 \times 0.70$

Table 1: Test datasets.

Model	Shell offset (mm) (specified)	Min. distance (mm) with Binary DT	Min. distance (mm) with AA DT
Fibula cutting guide 1	2.0	1.26	1.57
Mandibular resection guide 1	0.5	-	0.26
Plate 1	0.5	0.06	0.17
Fibula cutting guide 2	2.0	1.15	1.51
Mandibular resection guide 2	0.5	-	0.04
Plate 2	0.5	0.26	0.30
Fibula cutting guide 3	2.0	1.41	1.63
Mandibular resection guide 3	0.5	-	0.25
Plate 3	0.5	0.21	0.28

Table 2: Distance measurements.

tion with clinicians, we decided to use a shell offset of 2.0 mm for the fibula cutting guides, to take the membrane around the bone of the fibula into account, and a shell offset of 0.5 mm for the plates and mandibular resection guides, to provide some tolerance for the choice of bone threshold value in the segmentation.

To quantitatively evaluate the accuracy and precision of the generated models, we exported STL files of iso-surfaces of bone templates and models and imported the files in ParaView [1]. We used a Hausdorf distance filter plugin for ParaView [4] to compute surface-to-surface distances between bone templates and models (Figure 6). Fibula cutting guides and plates were generated with binary and AA DTs (Figure 7), wheras mandibular resection guides had similar properties (same shell offset) as plates and were generated only with AA DT. Table 2 compare the expected shell offset with the minimum distance computed for each object. Using the AA DT resulted in minimum distances closer to the expected shell offsets, and visibly smoother iso-surfaces, compared to the binary DT. The low minimum distance for the mandibular resection guide in Case 2 was caused by two drill guide components being inserted too close to the bone.

To test the manufacturability, we also printed a number of exported STL files in polyactide (PLA) plastic on an Ultimaker Original 3D printer (Figure 8).

The experiments were performed on a laptop with an Intel Core i7-4710MQ CPU, 16 GB of RAM, and a NVIDIA Quadro K4100M GPU.

5 DISCUSSION

Surgical guide and plate models can be efficiently generated in a few minutes from segmented CT images with the methods we present in this paper. We are currently preparing a validation study in which we for a larger number of patient cases will compare the planned outcome from our software with the actual outcome of simulated surgery on 3D printed plastic bones and models created from the plan.

An issue of the voxel-based modeling is the limited resolution it provides. Using an anti-aliased DT allows us to preserve sub-voxel distances, when fuzzy coverage representations are available, but fine details such as threads for screw holes require higher resolution. Adaptive distance fields [8], or performing parts of the CSG operations on the final iso-surface mesh, could be viable options to explore.

The use of haptics is also something to explore further for this type of application. In particular, haptic feedback for testing the fit of models during the design could provide valuable information to the user, and will eventually be added to our software.

ACKNOWLEDGEMENT

We like to thank Johan Nysjö, Centre for Image Analysis, for providing access to the BoneSplit segmentation software. We would also like to thank Andreas Thor, Andrés Rodríguez-Lorenzo, and Jan-Michaél Hirsch from the Department of Surgical Sciences, Uppsala University, as well as Daniel Buchbinder from the Mount Sinai School of Medicine, NY, for their clinical input and for providing datasets for Cases 2 and 3.

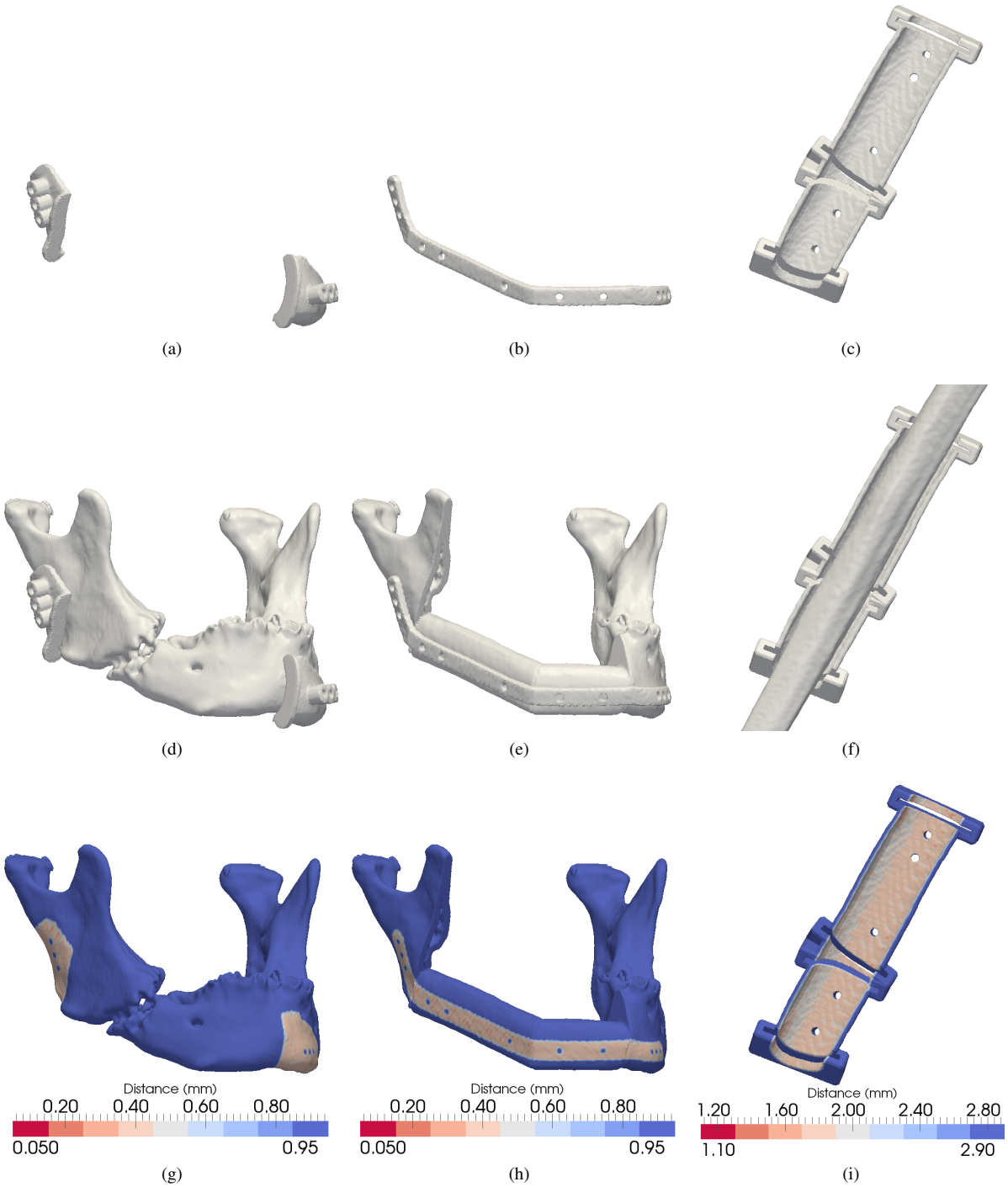
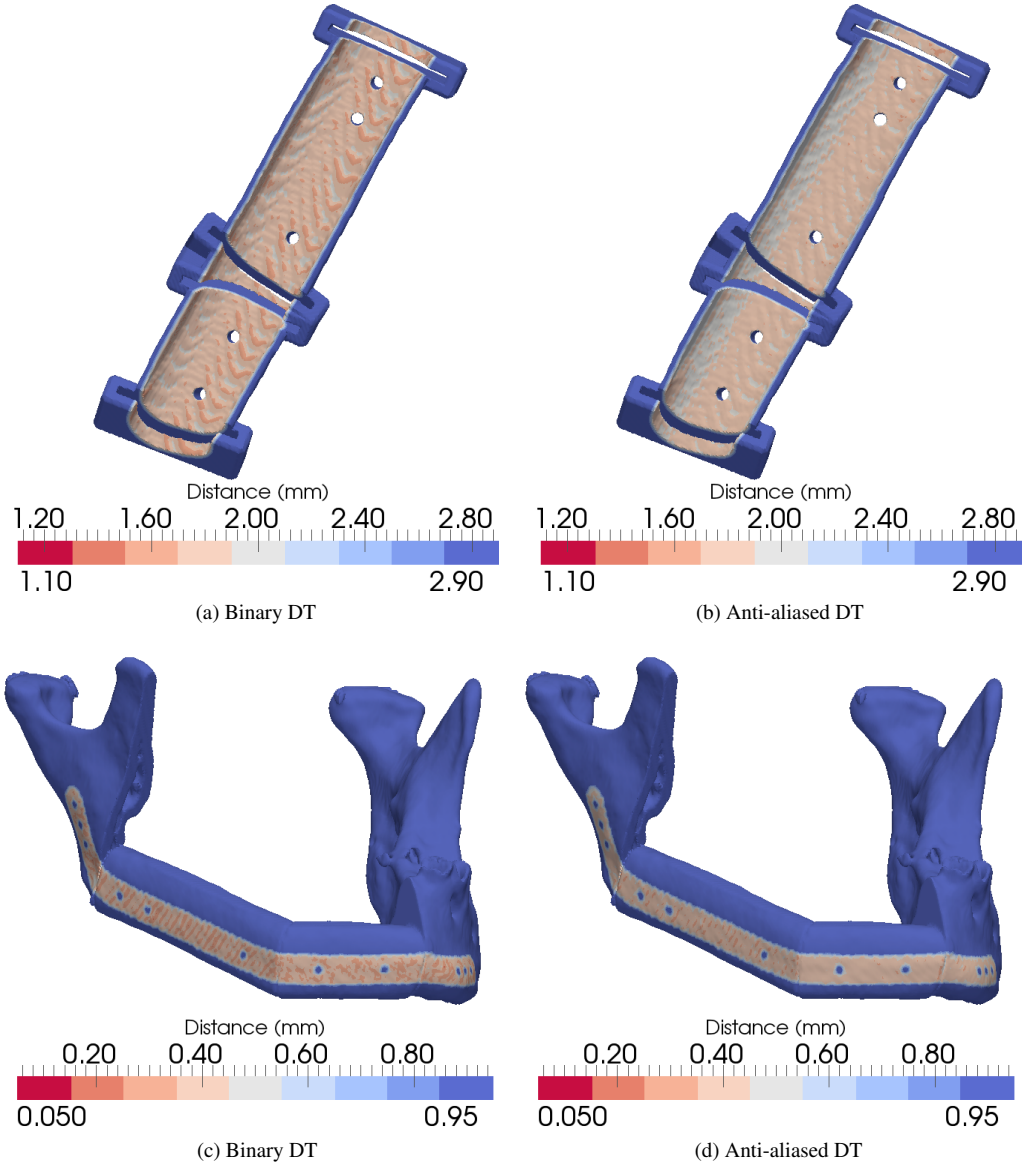


Figure 6: Results for Case 3: top row (a–c): iso-surfaces of generated guide and plate models; middle row (d–f): iso-surfaces of generated models with bone templates; bottom row (g–i): computed bone-to-model distances.



(a) Binary DT (b) Anti-aliased DT

(c) Binary DT (d) Anti-aliased DT

Figure 7: Comparison of bone-to-model distances for models generated with binary and anti-aliased DTs: (a–b) fibula cutting guide; (c–d) plate. This figure is best viewed on a monitor.

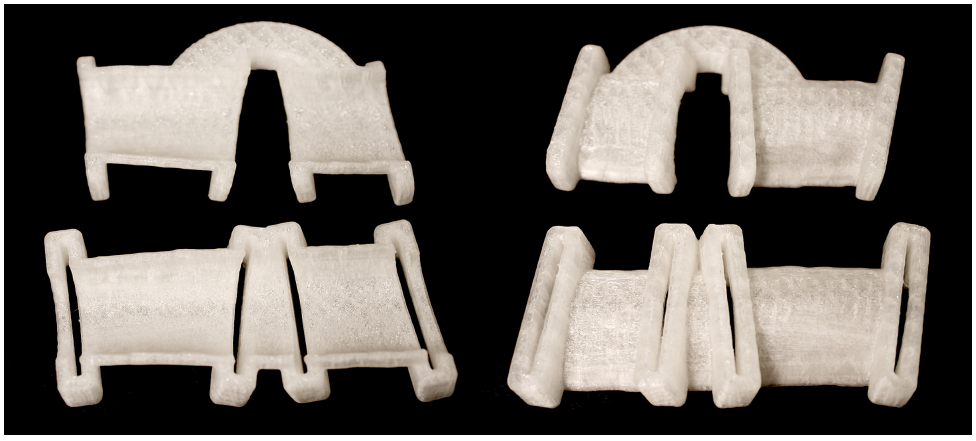


Figure 8: Fibula cutting guide models 3D printed in polyactide (PLA) plastic.

6 REFERENCES

[1] AHRENS, J., GEVECI, B., AND LAW, C. ParaView: An End-User Tool for Large Data Visualization, Visualization Handbook. Elsevier, 2005, ISBN-13: 978-0123875822.

[2] BORGEFORS, G. Distance Transformations in Arbitrary Dimensions. *Computer Vision, Graphics, and Image Processing 27*, 3 (1984), 321–345.

[3] BOURKE, P. Polygonising a scalar field. `http://paulbourke.net/geometry/polygonise`. Accessed on February 9, 2017.

[4] COMMANDEUR, F., VELUT, J., AND ACOSTA, O. A VTK Algorithm for the Computation of the Hausdorff Distance. *The VTK Journal* (2011). `http://hdl.handle.net/10380/3322`.

[5] DANIELSSON, P.-E. Euclidean Distance Mapping. *Computer Graphics and Image Processing 14*, 3 (1980), 227–248.

[6] FELZENSZWALB, P. F., AND HUTTENLOCHER, D. P. Distance Transforms of Sampled Functions. *Theory of Computing 8* (2012), 415–428.

[7] FORNARO, J., KEEL, M., HARDERS, M., MARINCEK, B., SZEKELY, G., AND FRAUENFELDER, T. An interactive surgical planning tool for acetabular fractures: initial results. *Journal of Orthopaedic Surgery and Research 50*, 5 (2010).

[8] FRISKEN, S. F., PERRY, R. N., ROCKWOOD, A. P., AND JONES, T. R. Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2000), SIGGRAPH '00, ACM, pp. 249–254.

[9] GEOMAGIC. Freeform. `http://www.geomagic.com/en/products/freeform/overview`. Accessed on February 9, 2017.

[10] GUSTAVSON, S., AND STRAND, R. Anti-aliased Euclidean distance transform. *Pattern Recognition Letters 32*, 2 (2011), 252–257.

[11] JONES, M. W., BAERENTZEN, J. A., AND SRAMEK, M. 3D Distance Fields: A Survey of Techniques and Applications. *IEEE Transactions on Visualization and Computer Graphics 12*, 4 (2006), 581–599.

[12] KOVLER, I., JOSKOWICZ, L., WEIL, Y., KHOURY, A., KRONMAN, A., MOSHEIFF, R., LIEBERGALL, M., AND SALAVARRIETA, J. Haptic computer-assisted patient-specific preoperative planning for orthopedic fractures surgery. *Int. Journal of Computer Assisted Radiology and Surgery 10*, 10 (2015), 1535–1546.

[13] LORENSEN, W. E., AND CLINE, H. E. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics 21*, 4 (1987), 163–169.

[14] MATERIALISE. Mimics. `http://www.materialise.com/en/medical/mimics-innovation-suite`. Accessed on February 9, 2017.

[15] MCNEELY, W. A., PUTERBAUGH, K. D., AND TROY, J. J. Six Degree-of-freedom Haptic Rendering Using Voxel Sampling. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM, pp. 401–408.

[16] MUSETH, K., BREEN, D. E., WHITAKER, R. T., AND BARR, A. H. Level set surface editing operators. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2002), SIGGRAPH '02, ACM, pp. 330–338.

[17] NYSJÖ, J., MALMBERG, F., SINTORN, I.-M., AND NYSTRÖM, I. BoneSplit - A 3D Texture Painting Tool for Interactive Bone Separation in CT Images. *Journal of WSCG 23*, 2 (2015), 157–166.

[18] OLSSON, P., NYSJÖ, F., HIRSCH, J.-M., AND CARLBOM, I. B. A Haptics-Assisted Cranio-Maxillofacial Surgery Planning System for Restoring Skeletal Anatomy in Complex Trauma Cases. *Int. Journal of Computer Assisted Radiology and Surgery 8*, 6 (2013), 887–894.

[19] OLSSON, P., NYSJÖ, F., RODRIGUEZ-LORENZO, A., THOR, A., HIRSCH, J.-M., AND CARLBOM, I. B. Haptics-assisted Virtual Planning of Bone, Soft Tissue, and Vessels in Fibula Osteocutaneous Free Flaps. *Plastic and Reconstructive Surgery - Global Open 3*, 8 (2015).

[20] OSIRIX. DICOM Image Library. `http://www.osirix-viewer.com/resources/dicom-image-library`. Accessed on February 9, 2017.

[21] PAYNE, B. A., AND TOGA, A. W. Distance field manipulation of surface models. *IEEE Computer Graphics and Applications 12*, 1 (1992), 65–71.

[22] QUILEZ, I. Distance functions. `http://www.iquilezles.org/www/articles/distfunctions/distfunctions.htm`. Accessed on February 9, 2017.

[23] RITACCO, L. E., MILANO, F. E., AND CHAO, E. *Computer-Assisted Musculoskeletal Surgery: Thinking and Executing in 3D*. Springer International Publishing, 2016.

[24] VOSS, J. O., VARJAS, V., RAGUSE, J.-D., THIEME, N., RICHARDS, R. G., AND KAMER, L. Computed tomography-based virtual fracture reduction techniques in bimandibular fractures. *Int. Journal of Cranio-Maxillofacial Surgery 44*, 2 (2015), 177–185.

[25] ZWEIFEL, D. F., SIMON, C., HOARAU, R., PASCHE, P., AND BROOME, M. Are Virtual Planning and Guided Surgery for Head and Neck Reconstruction Economically Viable? *Journal of Oral and Maxillofacial Surgery 73*, 1 (2015), 170–175.