AutoImplant 2020 - First MICCAI Challenge on Automatic Cranial Implant Design

Jianning Li, Pedro Pimentel, Angelika Szengel, Moritz Ehlke, Hans Lamecker, Stefan Zachow, Laura Estacio, Christian Doenitz, Heiko Ramm, Haochen Shi, Xiaojun Chen, Franco Matzkin, Virginia Newcombe, Enzo Ferrante, Yuan Jin, David G. Ellis, Michele R. Aizenberg, Oldřich Kodym, Michal Španěl, Adam Herout, James G. Mainprize, Zachary Fishman, Michael R. Hardisty, Amirhossein Bayat, Suprosanna Shit, Bomin Wang, Zhi Liu, Matthias Eder, Antonio Pepe, Christina Gsaxner, Victor Alves, Ulrike Zefferer, Gord von Campe, Karin Pistracher, Ute Schäfer, Dieter Schmalstieg, Bjoern H. Menze, Ben Glocker, and Jan Egger

***Abstract*— The aim of this paper is to provide a comprehensive overview of the MICCAI 2020 AutoImplant Challenge[1]. The approaches and publications submitted and accepted within the challenge will be summarized and reported, highlighting common algorithmic trends and algorithmic diversity. Furthermore, the evaluation results will be presented, compared and discussed in regard to the challenge aim: seeking for low cost, fast and fully automated solutions for cranial implant design. Based on feedback from collaborating neurosurgeons, this paper concludes by stating open issues and post-challenge requirements for intra-operative use. The codes can be found at https://github.com/Jianningli/tmi.**

***Index Terms*— Volumetric shape completion, Shape inpainting, Skull reconstruction, Shape prior, Statistical shape model, Deep learning, Cranioplasty.**

I. Introduction

CRANIOPLASTY is a reconstructive surgery to repair skull damages resulting from brain tumor surgeries or head trauma, where a part of the skull bone (mainly in the neurocranium area) has to be removed. Increased use of decompressive craniectomies resulted in more reconstructions of cranial defects in the past 15 years, around 25 patients per one million inhabitants per year for Europe, the Middle East and Africa [1], [2]. However, complications, like brain swelling and infections after decompressive craniectomies and cranioplasties, are frequent and can even be life-threatening events [3]. A systematic review revealed that one in 10 patients undergoing a decompressive craniectomy suffers a complication, which makes an additional medical or surgical intervention necessary [4]. Hence, a tailor-made patient-specific implant (PSI) of the cranium is needed in such surgery to optimally restore the protective, mechanical and anatomical functions of the human skull [5]. The design of a PSI remains a bottleneck [6] for cranioplasty, since the reconstructive surgery can be performed only after the implant has been designed, manufactured and delivered to the hospital, which may take weeks or even months. If cranioplasty could be performed immediately after the primary surgery that removes the skull bone, the overall duration of surgery can be reduced substantially. To achieve this goal, a fast, fully automatic and in-operating-room (in-OR) manufacturing of PSI is required. Additive manufacturing or 3D printing enables fast manufacturing of 3D medical implants directly in the surgery room, given the corresponding 3D models. Currently, the patient's head is scanned by computed tomography (CT) after primary surgery. The bone structures are extracted from the CT, converted into a 3D model and used to guide the computer-aided design (CAD) of the implant [7]–[10]. Symmetry is often assumed in CAD procedures, which use a mirrored copy of the healthy skull side as a template. However, symmetry cannot be used when the skull is deformed or when the defect crosses the symmetry plane.

Inspired by the clinical practice of relying on a post-operative head CT for cranial implant design, the AutoImplant 2020 challenge encouraged the development of automated implant design by providing both pre- and post-operative skulls for supervised training and evaluation. Unlike the clinical practice, which models implants as meshes, the challenge encouraged participants to predict the binary implant masks directly from binary skull images (voxel grids). Ten full papers were accepted by the challenge. They cover a variety of data-driven methods, including classical statistical approaches, such as statistical shape models (SSM) [11], and deep learning approaches, such as generative adversarial networks (GAN) [12], variational auto-encoders (VAE) [13] and variants of U-Net [14], which are novel in neurosurgery. From a technical perspective, the processing of high-dimensional skull data and the generalization to varied skull defects are key considerations for the development and evaluation of the algorithms.

This work was supported by CAMed (Clinical Additive Manufacturing for Medical Applications, COMET K-Project 871132, https://www.medunigraz.at/camed/), which is funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT), the Austrian Federal Ministry for Digital and Economic Affairs (BMDW) and the Styrian Business Promotion Agency (SFG). Further, this work received funding from the Austrian Science Fund (FWF) KLI 678-B31 and the TU Graz Lead Project (Mechanics, Modeling and Simulation of Aortic Dissection).

Author affiliations can be found on the Acknowledgment section.

[1]https://autoimplant.grand-challenge.org/

TABLE I: Quantitative results (mean DSC and HD) of the participating algorithms on $D_{test100}$ and D_{test10}.

Metrics\Alg	A1	A2.	A3	A3 (s)	A4	A5	A6	A7	A8	A8 (re)	A9 (r)	A9 (p)	A10 (r)	A10 (bbox)
DSC (100)	0.917	0.931	0.913	0.845	0.944	0.920	0.907	0.896	0.887	0.891	0.735	0.889	0.810	0.856
DSC (10)	0.919	0.924	0.769	0.816	0.932	0.910	0.870	–	0.351	0.473	–	–	–	–
HD (100)	4.336	3.660	4.067	6.414	3.564	4.137	4.180	4.602	7.017	6.909	7.243	5.534	5.440	5.183
HD (10)	3.987	4.090	8.585	5.952	3.934	4.707	4.760	–	29.476	21.049	–	–	–	–

II. Related Work

Prior to the challenge, automatic cranial implant design has been an under-researched area, especially concerning data-driven approaches, due to a lack of public datasets suitable for the task. This section summarizes the algorithms published online prior to the conclusion of the challenge, which have been used for automatic reconstruction of medical implants, including cranial implants. A review of general shape completion algorithms will also be covered in this section. An early study casts cranial implant design as a surface interpolation problem, smoothly interpolating the missing surface using radial basis functions [15].

A. Statistical Shape Model

Prior to the challenge, SSM is among the most widely used methods for reconstructing skull bones, including the facial area [16], [17], the cranium area [18] and other bone structures on the skull [19], [20]. A statistical model of the skull $\mathbf{S}(w)$ represents the average shape $\overline{S} \in \mathbf{R}^{3m}$ (m is the number of vertices of the skull mesh) as well as a set of shape variations $p_i \in \mathbf{R}^{3m}$ of a given skull population:

$$\mathbf{S}(w) = \overline{S} + \sum_{i=1} w_i p_i \tag{1}$$

Here, w_i is the shape weight of each mode of shape variation p_i, and its value is confined to the scope of the training skull population. Reconstructing a complete skull given a defective skull $\mathbf{D}$ is the task of finding the set of weight parameters w^* such that $\mathbf{S}(w^*)$ best matches the shape of $\mathbf{D}$, except in the defective region. The cranial implant can then be obtained by taking the difference (logical *XOR*) of the reconstructed skull and $\mathbf{D}$. Finding $\mathbf{S}(w^*)$ is usually an iterative process.

B. Deep Learning

Recently, deep learning solutions have emerged. Morais et al. [21] were the first to demonstrate a denoising auto-encoder for skull shape completion on very coarse skulls (dimension: 30^3, 60^3 and 120^3) with simple holes. Li et al. [6], [22] extended the concept of skull shape completion to high-resolution data, i.e., $512^2 \times Z$ with much irregular synthetic defects. Their approach showed potential for clinical use according to the evaluation results on real defective skulls from craniotomy. Their dataset is not yet public. Kodym et al. [23] trained a cascade of convolutional neural networks to predict the implants directly from synthetically defective skulls, using a publicly available dataset [2]. The trained model can also be generalized well to real head trauma-related defects, as the synthetic defects the authors created are closely mimicking real ones. The real defects used in this study are not publicly available. Matzkin et al. [24] focused on cranial implant design for decompressive craniectomy. The authors explored the possibility of both skull shape completion and predicting the implant directly from defective skulls using a U-Net style network. Similar to the studies described above, only synthetic defects are used for training, while, for evaluation, real cases are included. However, the dataset is not yet publicly accessible. These prior studies reveal that the algorithms, if carefully designed, can be generalized to real clinical defects, even if only synthetic defects are used during training. These prior algorithms also accept as input the 3D binary skull images (voxel grids) and produce the implants in the same format. However, the implants need to be converted to meshes in order to be 3D printed. The deep learning method by Zhang et al. [25] is focused on the maxilla area of the skull.

[2]https://www.fit.vut.cz/person/ikodym/skullbreak

C. Shape Completion

As earlier studies discussed in Section II. RELATED WORK (B) cast automatic cranial implant design as a volumetric shape completion problem, this section reviews the general shape completion algorithms used for various data modalities (points, meshes and voxel grids).

1) Voxel Grid Completion: Classical shape completion algorithms [26], [27] deal with volumetric data, which are voxelized from a point representation using a signed distance function. Voxel grid methods have been prevalent in recent studies using convolutional neural networks (CNN) on volumetric images. Both works employ an encoder-decoder style network, which is however restricted to accept as input coarse voxel grids (e.g., 32^3). Meshes can be extracted from the final completed grids.

2) Point/Mesh Completion: Recent development in deep learning enable a CNN to learn from unstructured point clouds efficiently. An encoder-decoder can be used to perform shape completion directly on the raw point data [28], [29] derived from Shapenet [30], which is often used as a benchmark dataset for both the voxel grid and point-based shape completion studies. Liu et al. [31] propose a two-step approach to complete dense point clouds. The first step predicts a completed but coarse point cloud using an encoder-decoder style network. In the second step, a residual network is used to produce a dense (high-fidelity) version of the completed point cloud, given as input a combination of the coarse output from the previous step and the partial point cloud. Early studies from Liepa (2003) [32] and Kraevoy et al. (2005) [33] perform shape completion directly on triangular meshes using classical geometry processing and mesh editing techniques.

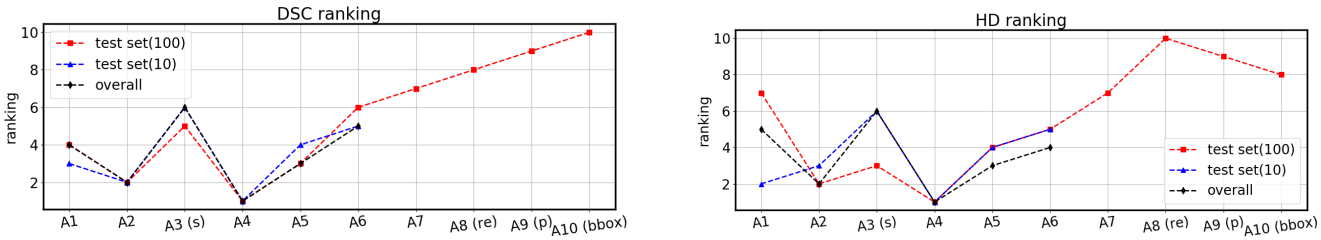


Fig. 1: DSC and HD Rankings of the Algorithms $A1$-$A10$, on $D_{test100}$, D_{test10} and the overall test set.

Shape completion on triangular meshes tend to be much more complicated than on binary voxel grids, as the former data structure can carry much richer information (e.g, texture, color) of an object compared to the latter.

3) Medical Images: Shape completion has also been applied to medical images[3]. Prutsch et al. [34] used a GAN to complete 2D aortic dissection (AD) images (CT), in order to generate the healthy aorta images prior to dissection. Armanious et al. [35] trained a GAN style network to complete arbitrarily shaped regions on 2D brain images. Gapon et al. [36] adopted a patch similarity matching method to remove metal artifacts on 2D CT and MRI images. A multi-layer perceptron (MLP) was trained to search the best matching patches to the missing region across an image. Manjón et al. [37] used a 3D U-Net to remove lesions on brain MRI images. The trained network can complete the missing region without requiring manual delineation of the lesions, while other studies all require explicit definition of the region of interest (usually done manually) before completion.

It should be noted that these medical shape completion applications require the restoration of not only the shape but also the voxel/pixel intensities of the missing region, as medical images are usually gray-scale. However, for the skull shape completion task in our challenge, we consider primarily the restoration of the missing shapes as the skull data are binary, containing only 0 and 1.

III. The AutoImplant Challenge

A. Organization, Evaluation and Ranking

The challenge was organized as a satellite event in MICCAI 2020, held virtually due to the COVID-19 pandemic. To our knowledge, this is also the first public challenge targeting the automatic design of cranial implants. Ten teams submitted their prediction results valid for evaluation, along with ten full papers. For ease of reference, the algorithms in the ten papers are denoted as $A1$ [38], $A2$ [39], $A3$ [40], $A3$ (s) [40], $A4$ [41], $A5$ [42], $A6$ [43], $A7$ [44], $A8$ [45], $A8$ (re) [45], $A9$ (r) [46], $A9$ (p) [46], $A10$ (r) [47] and $A10$ (bbox) [47], respectively. Among the algorithms, some [40], [45] have reported an enhanced version of their algorithm, denoted as $A3$ (s) and $A8$ (re), besides the base implementation $A3$ and $A8$. Two papers [46], [47] reported approaches for comparison, denoted as $A9$ (r), $A9$ (p) and $A10$ (r), $A10$ (bbox).

Two metrics, Dice Similarity Coefficient (DSC) and symmetric Hausdorff distance (HD, measured in millimeter) are used for quantitative evaluation of the results. DSC and HD are first ranked separately in descending order and ascending order, respectively, and the final ranking is obtained by taking the average of the two rankings, as shown in Figure 1. For [40], [45], [46] and [47], $A3$ (s), $A8$ (re), $A9$ (p) and $A10$ (bbox) are used for ranking. Table I shows the quantitative results (mean DCS and HD) of each algorithm. To get the results, participants needed to submit the predicted implants to the organizers, and a $.csv$ file containing the DSC and HD of each test case is returned to them. It was required that the predicted implants are of the same dimension as the corresponding defective skulls, i.e., $512^2 \times Z$ to be considered as valid submissions.

Table II shows the t-test for DSC and HD on the entire test set ($D_{test100}$ and D_{test10} combined together) among the leading methods ($A4$, $A2$, $A1$, $A5$, $A6$, $A3$ (s)). We can see that most of the p values are far smaller than 0.05, indicating that the differences among these leading methods are statistically significant. In particular, the winning method ($A4$) can beat its followers by a large margin, statistically speaking. In contrast, the difference between $A1$ and $A5$ is not significant for both DSC and HD. We also show the t-test between some network variants, i.e., $A3$ (s) $\leftrightarrow$ $A3$, $A8$ (re) $\leftrightarrow$ $A8$, $A9$ (p) $\leftrightarrow$ $A9$ (r) and $A10$ (bbox) $\leftrightarrow$ $A10$ (r) in Table II. Except $A9$ (p) $\leftrightarrow$ $A9$ (r) and $A10$ (bbox) $\leftrightarrow$ $A10$ (r), the t-test is run on the entire test set.

B. Challenge Dataset

We included a data descriptor [48] of the challenge dataset in the challenge proceedings, which detailed the origin, creation and statistics of the challenge dataset. However, a brief description of the training set and test set is provided in this section to make the contribution self-contained. We use the term *complete skull* to refer to the undamaged skull and *defective skull* to refer to a skull with a defect. The complete skulls in the challenge dataset are segmented from a public head CT collection CQ500 (http://headctstudy.qure.ai/dataset), using a thresholding technique (150 Hounsfield Units). The defective skulls are created automatically by removing part of the skull bone from the complete skulls.

1) Training Set: The training set contains 100 complete skulls from different head CT scans and their corresponding synthetic defective skulls and implants. An implant is simply the logical *XOR* of the corresponding complete skull and defective skull. The defects in the training set follow a similar pattern as illustrated in Figure 2 (A), regarding the size, shape

[3]In these studies, *inpainting* is more commonly used than *completion*.

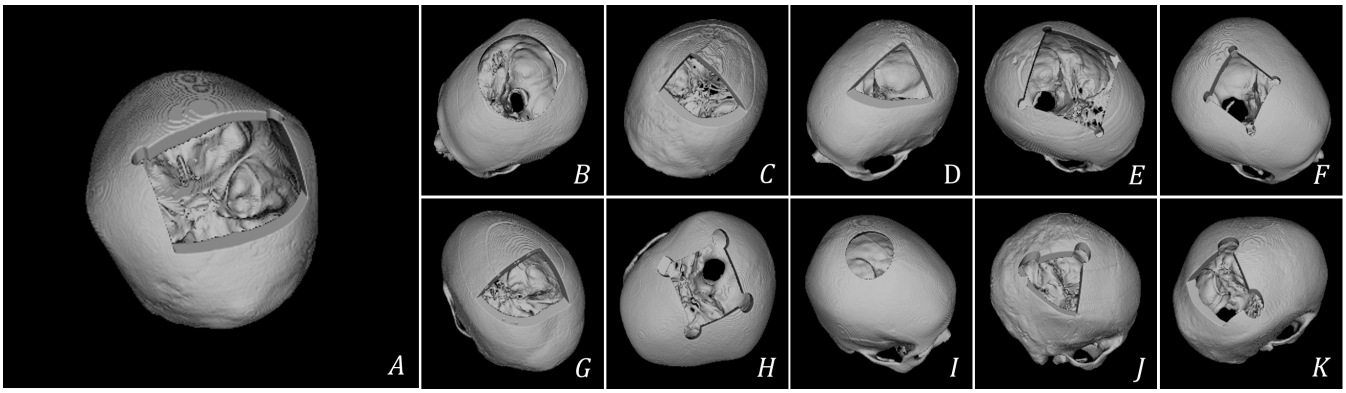


Fig. 2: Illustration of the skull defects in $D_{test100}$ and D_{test10}. The defect in (A) is representative of the defects in the training set and $D_{test100}$, where the defects are similar in terms of shape, size and position. (B)-(K) show the defects in D_{test10}, where there are three types of defects: spherical (B,I), cubic (C, D, G) and cubic with cylinders on the corners (E, F, H, J, K).

TABLE II: t-test between the top ranking methods ($A4$, $A2$, $A1$, $A5$, $A6$, $A3$ (s)) and some network variants for DSC and HD. p values larger than 0.05 ($5e^{-2}$) are highlighted.

	DSC	HD
$A4 \leftrightarrow A2$	$3.1e^{-4}$	$\mathbf{5.8e^{-1}}$
$A4 \leftrightarrow A1$	$2.4e^{-10}$	$3.0e^{-3}$
$A4 \leftrightarrow A5$	$6.7e^{-8}$	$1.2e^{-2}$
$A4 \leftrightarrow A6$	$9.6e^{-13}$	$2.3e^{-3}$
$A4 \leftrightarrow A3$ (s)	$1.4e^{-16}$	$1.3e^{-3}$
$A2 \leftrightarrow A1$	$2.3e^{-3}$	$9.5e^{-3}$
$A2 \leftrightarrow A5$	$1.5e^{-2}$	$3.4e^{-2}$
$A2 \leftrightarrow A6$	$1.2e^{-6}$	$8.5e^{-3}$
$A2 \leftrightarrow A3$ (s)	$1.3e^{-13}$	$1.8e^{-3}$
$A1 \leftrightarrow A5$	$\mathbf{7.4e^{-1}}$	$\mathbf{6.7e^{-1}}$
$A1 \leftrightarrow A6$	$1.2e^{-2}$	$\mathbf{7.8e^{-1}}$
$A1 \leftrightarrow A3$ (s)	$9.7e^{-11}$	$1.7e^{-2}$
$A5 \leftrightarrow A6$	$8.0e^{-3}$	$\mathbf{8.6e^{-1}}$
$A5 \leftrightarrow A3$ (s)	$6.4e^{-11}$	$1.1e^{-2}$
$A6 \leftrightarrow A3$ (s)	$1.8e^{-7}$	$1.3e^{-2}$
$A3$ (s) $\leftrightarrow A3$	$3.4e^{-6}$	$3.0e^{-2}$
$A8$ (re) $\leftrightarrow A8$	$\mathbf{5.3e^{-1}}$	$\mathbf{7.3e^{-1}}$
$A9$ (p) $\leftrightarrow A9$ (r)	$1.7e^{-47}$	$1.2e^{-3}$
$A10$ (bbox) $\leftrightarrow A10$ (r)	$5.9e^{-8}$	$\mathbf{4.5e^{-1}}$

and position. Participants were free to create and use additional defects on the complete skulls provided for training.

2) Test Set: Two independent test sets, denoted as $D_{test100}$ and D_{test10}, were created for evaluation of the submissions. $D_{test100}$ contains 100 defective skulls (created out of 100 skulls different from those in the training set), with defects in $D_{test100}$ similar to those in the training set (Figure 2 (A)). D_{test10} contains 10 defective skulls with varying defects, as shown in Figure 2 (B)-(K). Figures 1, 5 and 7 denote the two test sets as (100) and (10). We created the two test sets to evaluate the generalization performance of participants' algorithms. Considering that the skull shape is patient-specific and the shape of the defects from craniotomy also depends on the specific pathological conditions e.g., the size and position of the brain tumor, of the patients, we expect the participants' algorithms to generalize well across different skulls and defects, which are desired in cranioplasty. According to [48], the defect variation in D_{test10} is much greater than that in $D_{test100}$, which is primarily used to evaluate how well the algorithms can generalize across varied skull shapes, while D_{test10} evaluates whether the algorithms can generalize to randomly shaped, sized and positioned defects, especially when trained only on the training set with a fixed defect pattern shown in Figure 2 (A).

Algorithms $A1$ - $A6$ succeeded on both test sets, while $A7$ - $A10$ failed on D_{test10}, and thus $A7$ - $A10$ are not included in the ranking on D_{test10}, as shown in Figure 1. We also calculated the ranking of $A1$ - $A6$ on the entire test set ($D_{test100}$ and D_{test10} combined together).

All the images in the challenge dataset have dimension $512^2 \times Z$. The ground truth of the test set, i.e., the corresponding complete skulls and implants, were kept secret by the organizers.

3) Rationale: As introduced above, synthetic defective skulls are used in both the training and evaluation phase of our challenge. However, the synthetic defects, as shown in Figure 2 (A), are created to resemble the real craniotomy defects by including the drilling holes on the defect borders. In craniotomy, a cranial drill is used by neurosurgeons for drilling small roundish holes on human skulls in order to create an opening in the skull. A craniotome is further used to remove a bone flap to access the brain underneath. This course of action can result in a skull defect with small roundish corners, similar to the artificial ones used in our challenge. The drilling holes are also important for the insertion and fixation of a cranial implant in cranioplasty. However, real craniotomy defects tend to have rough boundaries, as the skull is cut manually using the craniotome, in contrast to the synthetic defects which have smooth and straight boundaries. The rationale for using synthetic defects in our challenge is twofold: First, as discussed in Section II. RELATED WORK (B), the algorithms can generalize to real craniotomy defects even if only synthetic defects are involved in the training phase. Second, using real defects in our current challenge is neither practical (not enough data and privacy restrictions) nor efficient (expert evaluations are needed due to a lack of ground truth for the real defects) especially when dozens of submissions are to be expected.

TABLE III: Summary and comparison of the algorithms.

Algorithm	Architecture	Input Dim	Hardware	Skull Preprocessing	Defect Augmentation	Use of Shape Prior	Output	D_{test10}	# Param
$A1$ [38]	SSM + 2D GAN	256×256	$4\times$ RTX 6000	yes	no	yes	skull	yes	229.18M
$A2$ [39]	ED + SE block	512×512	GTX 1080+GTX 960	yes	yes	no	implant	yes	3.17M
$A3$ [40]	U-Net	$304 \times 304 \times 224$	TITAN Xp	yes	yes	no	implant	yes	6.77M
$A3$ (s) [40]	U-Net + shape prior	$304 \times 304 \times 224$	TITAN Xp	yes	yes	yes	implant	yes	5.19M
$A4$ [41]	U-Net with residual block	$176 \times 224 \times 144$	$2\times$ V100	yes	yes	no	skull	yes	68.56M
$A5$ [42]	Cascade U-Net	$128 \times 128 \times 128$	Titan Xp	yes	yes	no	implant	yes	5.96M
$A6$ [43]	U-Net	$192 \times 256 \times 128$	RTX Titan	yes	yes	no	skull	yes	6.49M
$A7$ [44]	ED with residual block	$180 \times 180 \times 180$	Quadro P6000	no	no	no	skull	no	1.49M
$A8$ [45]	RDU-Net	$128 \times 128 \times 64$	$3\times$ RTX 2080 Ti	no	no	no	skull	no	2.51M
$A8$ (re) [45]	RDU-Net + VAE	$128 \times 128 \times 64$	$3\times$ RTX 2080 Ti	no	no	yes	skull	partially	25.46M
$A9$ (r) [46]	V-Net + resizing	$256 \times 256 \times 64$	RTX Titan	no	no	no	skull	no	45.60M
$A9$ (p) [46]	V-Net + patch	$256 \times 256 \times 64$	RTX Titan	no	no	no	implant	no	45.60M
$A10$ (r) [47]	ED + resizing	$128 \times 128 \times 64$	GTX 1070 Ti	no	no	no	implant	no	82.08M
$A10$ (bbox) [47]	ED + boundingbox	$256 \times 256 \times 128$	GTX 1070 Ti	no	no	no	implant	no	0.65M

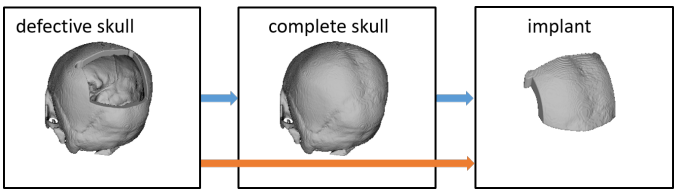


Fig. 3: Two types of problem formulation used among the submitted algorithms. The blue arrow indicates that the algorithms reconstruct a complete skull first, and then the implant is obtained through the subtraction of the defective skull from the reconstructed skull, which defines a *shape completion* problem. The orange arrow indicates that the algorithms reconstruct the implant directly from a defective skull.

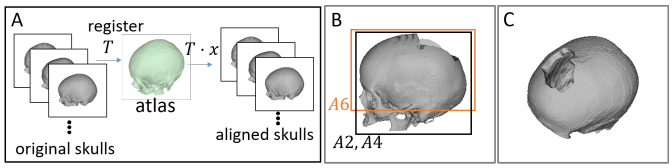


Fig. 4: Illustration of the skull preprocessing methods. (A) Aligning the skulls to a common skull atlas ($A3$). (B) Image background cropping ($A2$, $A4$, $A6$) and (C) Aligning the four anatomical landmarks on each skull onto a common axial plane ($A5$).

IV. Summary and Comparison of the Algorithms

This section summarizes the algorithms from the perspective of problem formulation, skull pre-processing, defect augmentation, network architecture, post-processing and skull dimension. Emphasis will be placed on how the submitted algorithms deal with high-dimensional skull data and on the generalization performance of these algorithms to highly varied skull defects. Table III provides a summary. Specific details of the algorithms can be found in the proceedings [49].

A. Problem formulation

As illustrated in Figure 3, two types of problem formulation are used among the algorithms: (1) Some participants formulated the problem of cranial implant design as a volumetric *shape completion* task. In this formulation, a complete skull is first reconstructed from a defective skull, and the implant is viewed as the difference between the complete skull and the defective skull. (2) Others view the problem as a shape learning task, and the shape of a implant is learned directly from the shape of a defective skull. The *Output* column in Table III shows the formulation adopted by each algorithm.

B. Preprocessing of the Skull

Preprocessing the skulls by removing the rotation, translation and other image-related variations helps the deep neural networks to focus on learning the shape variations of the skulls and defects, which is the primary concern of the challenge. This course of action also reduces the difference between the training and test set and thus can potentially help improve the final results. The commonly used techniques for this purpose include image background cropping [39], [41], [43] and skull alignment via registration [40], [42]. Table IV shows a description of the skull preprocessing techniques (if used) per algorithm.

1) Background Cropping: Zero-valued background voxels outside the skull's bounding box provide no useful information for shape learning. Cropping the background reduces image size and thus the memory consumption. In practice, instead of cropping the entire background, some margins are usually kept (Figure 4, B). After cropping, $A2$ further resized the cropped images to 512^3 so that the 2D slices of axial, sagittal and coronal planes have the same size (512^2). $A6$ cropped bone structures below the skull base irrelevant to the task, e.g., mandibles (Figure 4, B, $A6$). The cropped images were then downsampled to $192 \times 256 \times 128$ to get a fixed input size for the shape completion network. $A4$ downsampled the cropped images to $176 \times 224 \times 144$. Note that downsampling the original image volume ($512^2 \times Z$) directly to such low size

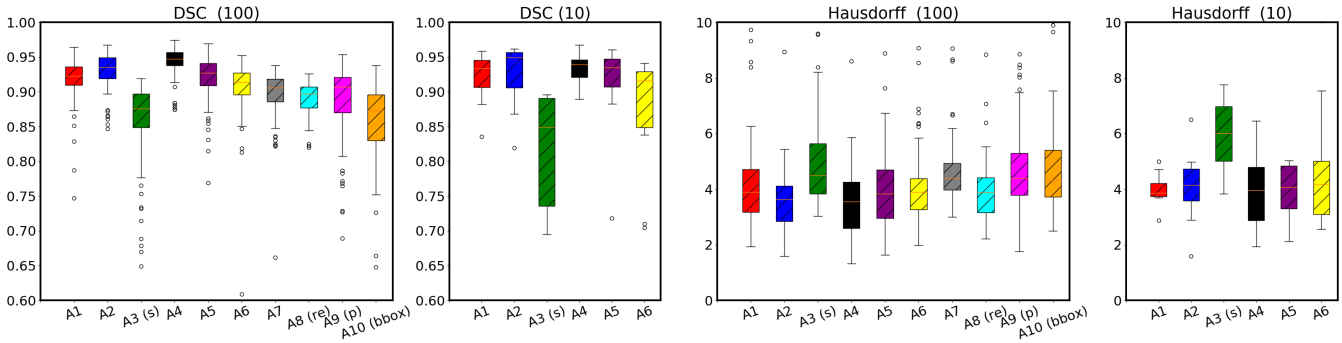
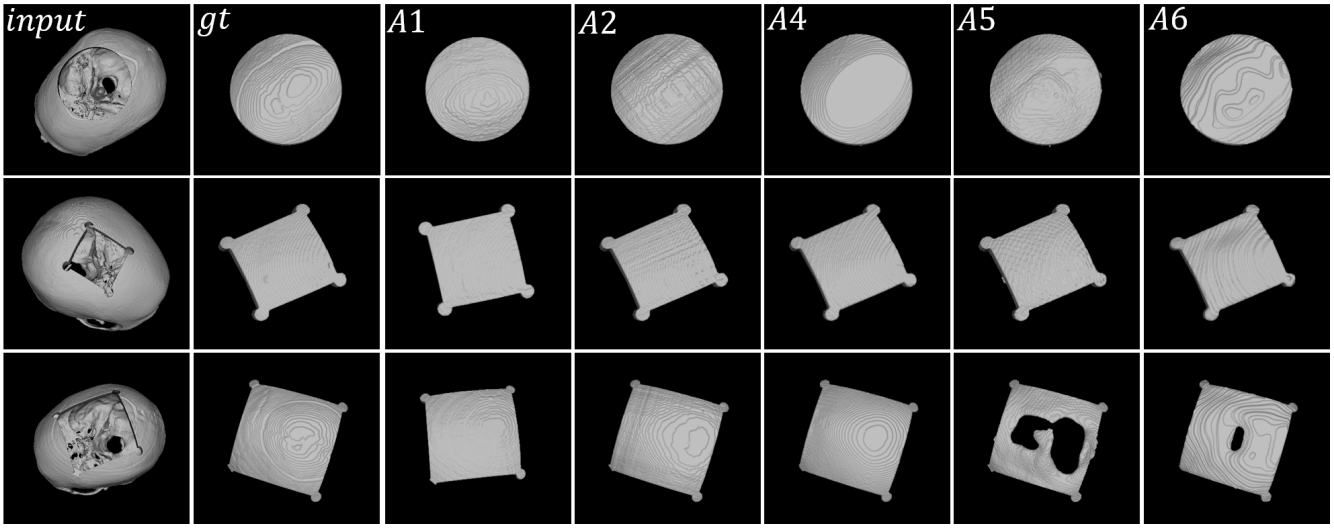
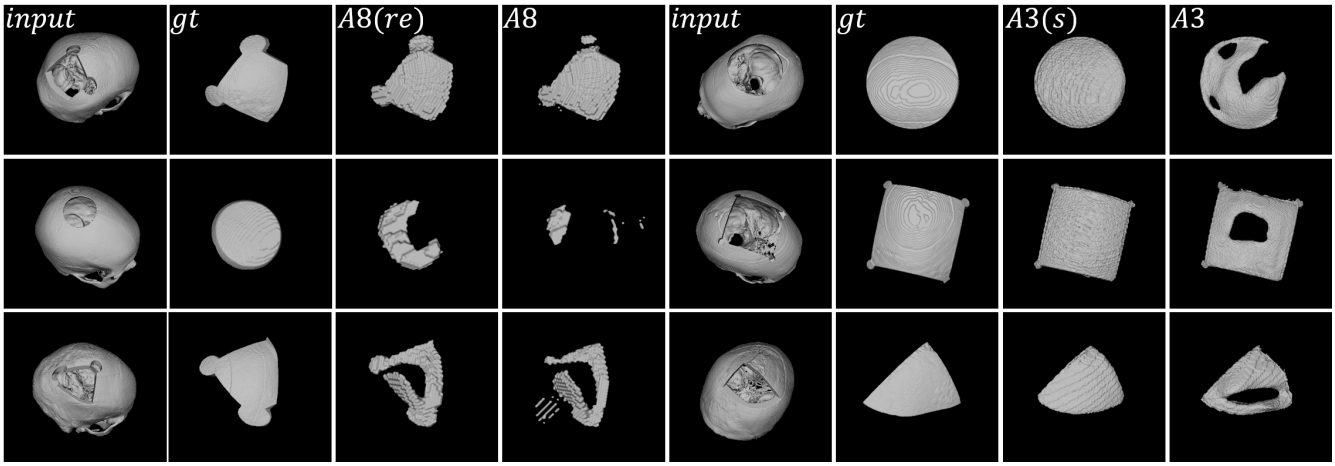


Fig. 5: DSC and HD of algorithms $A1$-$A10$ on $D_{test100}$ (100) and D_{test10} (10). Among the algorithms, $A7$-$A10$ failed on D_{test10} (10).



(a) Implant predictions from $A1$, $A2$, $A4$, $A5$ and $A6$ on D_{test10}.



(b) Comparison between the implant produced by $A8$ and $A8$ (re), $A3$ and $A3$ (s) on D_{test10}.

Fig. 6: Implant predictions on D_{test10}. (a) First to last column: the input, ground truth, predictions from $A1$, $A2$, $A4$, $A5$ and $A6$. (b) The first and fifth column show the input. The second and sixth column show the ground truth. The third and seventh column show the predictions from $A8$ (re) and $A3$ (s). The fourth and eighth column show the predictions from $A8$ and $A3$.

TABLE IV: Description of the skull preprocessing methods used by participants.

Algorithms	Preprocessing Methods
$A2$ [39]	Image background cropping and resizing the skull region to 512^3, so that the axial, sagittal and coronal slices have the same dimension 512^2.
$A3$ [40]	Align the training/test set to a common skull atlas via rigid registration.
$A4$ [41]	Image background cropping and image re-orientation
$A5$ [42]	Align the skull (on the x/y plane) based on four skull landmarks using rigid registration.
$A6$ [43]	Crop the image background and the area below the skull base and re-scale all images to 192×256×128 non-isotropically (different scaling factors in the $x-$, $y-$ and $z-$ direction).

tends to lead to considerable degradation of image quality and compromises the algorithmic performance. Cropping before downsampling can mitigate such adverse effects.

2) Skull Alignment: To reduce the rotational and translation variations, $A3$ and $A3$ (s) aligned all the skulls in the training and test set to a common skull atlas using rigid registration, as shown in Figure 4 (A). The skull atlas is constructed by averaging the shapes of several complete skulls. Such transformation also resamples the images to an intermediate size of $304^2 \times 224$. In $A5$, instead of using a pre-defined skull atlas, the alignment is based on four anatomical landmarks on the skulls, i.e., the left and right auditory meatus and left and right supraorbital notch. These landmarks are aligned onto the same axial plane using a rigid transformation. A U-Net style network is trained to detect the four landmarks automatically on the test set. Another benefit of such a transformation is that the unwanted bone structures below the alignment plane (e.g., midface, mandibles) can be discarded automatically (Figure 4, C). The *Skull Preprocessing* column in Table III shows whether the algorithms used a preprocessing step. As shown in Figure 5, algorithms that used skull preprocessing generally outperform those that did not. An ablation study performed for [42] demonstrated that using skull alignment improved the quantitative results on a validation set. However, this course of action adds another dimension of complexity. For example, in order to obtain the final prediction y given a test case x, $A3$ needs to consider the transformation matrix from registration $\mathbf{T}$ and its inverse $\mathbf{T}^{-1}$,

$$y = \mathbf{T}^{-1} \cdot f(\mathbf{T} \cdot x), \tag{2}$$

where f represents the deep neural network. Furthermore, such transformation can usually resample the images to a smaller size [40], which reduces the memory needed to process the skull data. Note that $A4$ re-oriented all the images to Right, Anterior, Superior (RAS), which has a similar effect to aligning the skulls in that the data are submitted to the U-Net in the same orientation. As a bonus, re-orientation can be done at runtime and does not require a registration.

C. Defect Augmentation

Among the submitted algorithms, augmentation of skull defects was a dominant factor contributing to the generalization of the algorithms to highly varied defects in D_{test10}. The defects in the training set have limited variations regarding the shape, size, and position, while the defects in the D_{test10} are of much greater irregularity. It is therefore challenging for the algorithms to generalize well to D_{test10} without the creation and use of additional defects during training, if a standard network configuration is used. In Table III, the *Defect Augmentation* column shows whether the algorithms have created and used additional defects for training besides those provided in the original challenge dataset. The D_{test10} column shows whether the algorithms can generalize to D_{test10}. We can see that algorithms that generalize to D_{test10} generally also used defect augmentation, with the exception of $A1$, which used a strong shape prior to guide the skull reconstruction process. Table V summarizes the defect augmentation techniques (if any) of the algorithms, which can be classified into three groups: (1) Create defects similar to those in $D_{test100}$ and D_{test10} shown in Figure 2 ($A2$, $A3$, and $A6$), (2) create random defects without resemblance of the defects in the test sets ($A5$), (3) create additional defective skulls using pair-wise registration and warping ($A4$).

1) Creating Defects Resembling the Test Set: $A2$ generated additional defects using a rectangular mask in axial, sagittal and coronal slices. To generate defects similar to $D_{test100}$ and D_{test10}, the rectangular mask was tailored according to the defect distributions in the respective test set. $A3$ and $A6$ generate defects directly on 3D volumes using 3D spherical and cubic masks. $A3$ created a mask combining cubes with cylinders to generate defects similar to the defects illustrated in Figure 2 (E, F, H, J, K). These augmentation strategies seek to create similar defect distributions on the training set to those of D_{test10}, which further allows the algorithms to generalize to D_{test10}. As introduced in Section III *B. Challenge Dataset*, for evaluation, only synthetic defects were used, which were similar but simplified compared to real craniotomy defects. The success of these augmentation strategies implies that creating synthetic defects that are closely resembling the real defects (craniotomy, traumatic brain injury or TBI, etc.) for training might help to increase the success rates of the algorithms in clinical scenarios.

2) Creating Random Defects: As shown in Figure 4 (C), instead of trying to generate defects similar to those of the test set, $A5$ created five defects with random shape, size and position on each skull, resulting in a total of $90 \times 5 = 450$ training pairs (10 skulls in the training set were reserved as a validation set). An ablation study revealed that the algorithm $A5$ can generalize to these random defects only if these augmented defects are also involved in the training phase.

3) Augmentation via Registration and Space Warping: D. G. Ellis et al. [41] augment the dataset by registering each skull in the training set with the remaining 99 skulls. For each registration, each skull can be warped into the space of the remaining 99 skulls using the corresponding transformation, yielding 99 uniquely warped skulls. This course of action substantially increases the number of training pairs to $99 \times 100 = 9900$,

TABLE V: Description of the data augmentation methods used by participants.

Algorithms	Augmentation Methods
$A2$ [39]	Creating defects on 2D slices in axial, sagittal and coronal planes using a rectangular mask.
$A3$ [40]	Creating 3D defects using random sized masks similar to the defects in D_{test10} (spherical, cubic, cube-cylinder).
$A4$ [41]	Permutation, scaling, translation and pair-wise non-linear registration and warping.
$A5$ [42]	Random lateral flipping and creating five random defects per skull.
$A6$ [43]	Creating defects using spherical and cubic masks.

excluding the defective skulls in the challenge dataset. $A4$ used a total of 9803 pairs for training (197 registrations failed).

The three defect augmentation strategies all have proven to be effective in improving the generalization performance of the algorithms on D_{test10}, as illustrated in Figure 6. For training, two algorithms [39], [40] intentionally created defects similar to the test sets, so that the algorithms can naturally generalize well to both of the test sets. However, even if $A6$ only augmented spherical and cubic defects, it can still generalize well to the defect pattern shown in the second row of Figure 6 (a). Similarly, $A5$ augmented random defects, but can generalize to other defect patterns (e.g., spherical defects), according to the first two rows in Figure 6 (a). The third row shows that $A5$ and $A6$ tend to perform worse on large defects. For $A4$, the good generalization performance can be largely attributed to the massive augmentation enabled by warping each training skull into the space of the remaining training cases. Unlike other augmentation techniques, which only try to increase the variations of the defects, while the shape variations of the skull are limited to the original training set, this course of action essentially created new skulls.

D. Architecture and Network Configurations

The *Architecture* column in Table III lists the deep learning models upon which the algorithms are built. We can see that most algorithms, $A2$, $A7$, $A10$ (r) and $A10$ (bbox), are based on an encoder-decoder (ED) architecture or ED with skip connections, i.e., U-Net for $A3$, $A3$ (s), $A4$, $A5$, $A6$ and $A8$. For $A1$, the primary part of the algorithm is based on a statistical shape model (SSM), which reconstructs a complete skull given a defective skull. A generative adversarial network (GAN) is further used to refine the output of the SSM. The GAN is trained using 2D slices from complete skulls from the training set. The generator component of the GAN is an auto-encoder network, which is trained to generate refined 2D skull slices. During the inference stage, the generator takes as input a combination of 2D slices from the test case (defective skull) and the complete skull reconstructed by the previous SSM, and produces the completed 2D slices, which are aggregated to form the final complete skull in 3D. The corresponding implant is obtained by subtracting the test case from the final reconstructed 3D complete skull. $A3$, $A6$, $A9$ and Li et al. [47] used standard U-Net, V-Net or ED configurations, while variants of ED and U-Net were explored by the other algorithms.

$A2$ uses a Squeeze-and-Excitation (SE) block [50], which introduces channel attention mechanisms, and an auxiliary path formed by several convolutional layers, to connect the encoder and decoder part of the network. $A2$ uses a standard U-Net to directly predict the implants from defective skulls. However, $A3$ proposed a way to incorporate shape prior of the skull into the network, which aims to improve the generalization ability of the network. The base algorithm and the shape-prior-enhanced version are denoted as $A3$ and $A3$ (s), respectively. $A4$ used a U-Net with residual blocks [51] in each level of the convolutional and deconvolutional layers. $A5$ used two U-Nets in a cascaded fashion; the first U-Net was responsible for producing a coarse implant of low resolution (128^3) given a downsampled defective skull as input, and the second U-Net was used as a *super-resolution* network to upsample the low-resolution prediction to high resolution, given as input a patch (128^3) of the prediction concatenated with the corresponding patch of the original high-resolution defective skull. Similar to $A5$, $A7$ also followed a two-step process to generate high-resolution predictions. First, a 3D ED was used to generate a low-resolution prediction of dimension 180^3. In the ED, residual blocks were used to connect the encoder part of the network with the decoder part. Second, a 2D decoder consisting of several convolutional layers with residual blocks and SE blocks was used to *super-resolve* the predictions to the original high resolution in a slice-wise manner. $A8$ used a Residual Dense U-Net (RDU-Net) [52] as the base implementation. The loss function of the network was enhanced using a shape regularization term derived from a pretrained variational auto-encoder network (VAE). We denote the network trained with and without the regularization term in the loss function as $A8$ and $A8$ (re). $A9$ (r) and $A9$ (p) were based on a V-Net architecture [53]. $A9$ (r) used a downsampled version of the skulls for training and produced low-resolution implants ($256^2 \times 64$), which were upsampled to the original dimension using simple image resizing. $A9$ (p) used a patch-based method to train on the original skull data using a V-Net. The base implementation of Li et al. [47] is $A10$ (r), which used a standard encoder-decoder to predict implant in low resolution ($128^2 \times 64$). Simple image resizing was used to upsample the implants to the original dimension. $A10$ (bbox) was built upon the output of $A10$ (r), which was used to extract a bounding box (bbox) on the original high-dimensional defective skulls. $A10$ (bbox) used another encoder-decoder network to predict high-resolution implants directly from the bounding box, which has a much smaller size than the original skulls. Most of the networks use DSC loss or a combination of DSC loss and cross-entropy loss as the objective function for this task.

The last column (# Param) of Table III shows the number of trainable parameters of the deep learning components of the algorithms. For $A1$, the number refers to the GAN. Note that the top-ranking algorithm $A4$ has significantly more parameters (68.56M) than its followers $A2$, which has only 3.17M parameters and $A5$, which has 5.96M parameters. $A10$

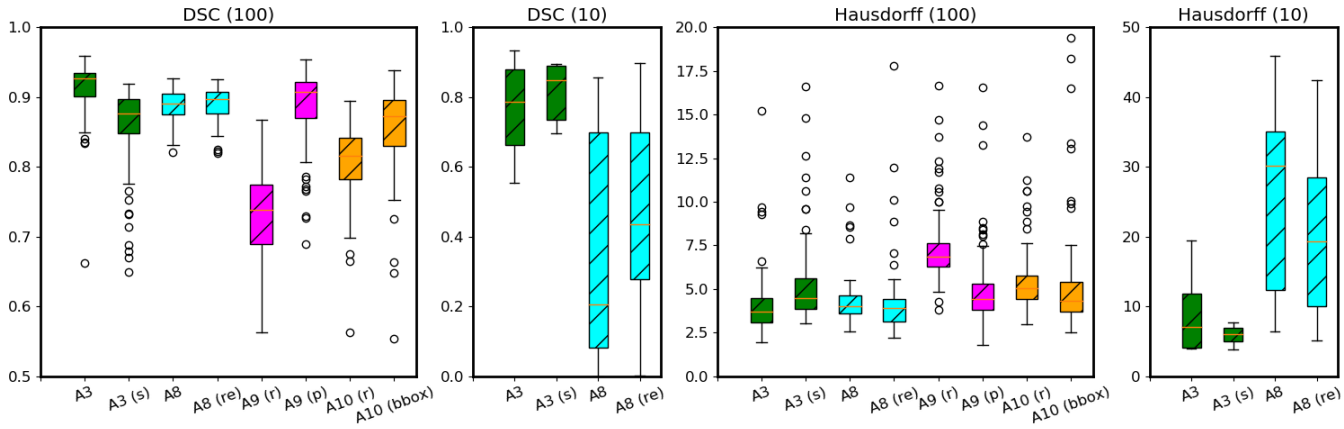


Fig. 7: Comparison between $A3$ and $A3$ (s), $A8$ and $A8$ (re), $A9$ (r) and $A9$ (p), $A10$ (r) and $A10$ (bbox) on $D_{test100}$ and D_{test10}.

(r), $A9$ (r) and $A9$ (p) have the second and fourth largest number of parameters, while their performances are ranked the last in our challenge.

E. Shape Priors

Besides defect augmentation, the exploitation of skull shape priors proves to be another effective measure to improve the generalization performance to varied defect patterns. The *Use of Shape Prior* column in Table III shows whether the skull shape priors is used by each algorithm. We see that even if algorithms $A1$ and $A8$ (re) used no augmented defects for training, skull shape priors let them still (partially) generalize to D_{test10}. Both quantitative and qualitative comparisons (Figure 7, Figure 6, b) demonstrate the advantages of shape priors, especially when it comes to defects different from the ones in the training set. The shape prior can be introduced either on-the-fly during the reconstruction process or during the learning process, using shape constraints or contextual information. Among the algorithms submitted, there are three different strategies for using the shape of a complete skull as prior knowledge: (1) Building a statistical model of the complete skulls ($A1$), (2) using the shape prior as contextual information during learning ($A3$), (3) using the shape prior as shape constraints in the loss function ($A8$).

1) Statistical Shape Model: A statistical shape model of the skull represents the *average* shape as well as principal shape variations of human skulls. The shape representation ability of a SSM is decided largely by the size and diversity of the skull dataset on which the SSM is built. $A1$ created a 3D skull SSM using the complete skulls from the training set, using principal component analysis (PCA). In the test phase, a defective skull is fitted to the SSM to find the shape variations that best match the shape of the given test case, during which the SSM acts as a strong shape prior to guide the skull reconstruction. The fitted shape serves as an initial approximation of the reconstructed complete skull corresponding to the test case and is further refined using a GAN. Note that, unlike other algorithms that used both defective skulls and complete skulls or the implant for training, the construction of the skull SSM and the training of the GAN only requires the complete skulls. Such *unsupervised* learning enables the algorithm to be independent from the defect patterns, and thus its performance is not affected by the shape, size and position of the defects. We can see from Figure 5 that it performs almost equally well on $D_{test100}$ and D_{test10}, even without augmenting the defects. Figure 6 (a) shows an illustration of the reconstruction results of $A1$.

2) Shape Prior as Contextual Information: $A3$ and $A3$ (s) are used to evaluate how the incorporation of shape prior affects the performance of the algorithm on D_{test10}. Both algorithm variants follow the same network and training configuration, except that $A3$ (s) uses a skull atlas as an additional input channel for the network during training. The atlas is the same as used for alignment in $A3$, which represents the average shape of several complete skulls. By doing so, the skull atlas can provide the contextual information beneficial for the learning process, which distracts the model from overfitting to the defect patterns in the training set and consequently improves robustness of the model. The ablation study of $A3$ shows that the algorithm performs better on D_{test10} when a shape prior is incorporated into the network, i.e., $A3$ (s) performs better than $A3$ regarding DSC and HD, as can be seen from the boxplot in Figure 7. According to Table II, the improvement of DSC and HD on the test set due to the introduction of the shape prior is statistically significant. Qualitatively, we can also see from Figure 6 (b) that $A3$ failed partially, whereas $A3$ (s) can succeed on some of the test cases from D_{test10}.

3) Shape Constraints in the Loss Function: B. Wang et al. [45] reported a comparison of a deep neural network trained with and without shape prior, denoted as $A8$ (re) and $A8$. In $A8$ (re), the shape prior is implemented as a regularization term in the loss function, which tries to minimize the Euclidean distance between the prediction and the ground truth in a latent feature space learned using a VAE. The VAE was trained end-to-end using the complete skulls in the training set to learn a compact and latent shape representation of the complete skulls. A RDU-Net was then used for skull

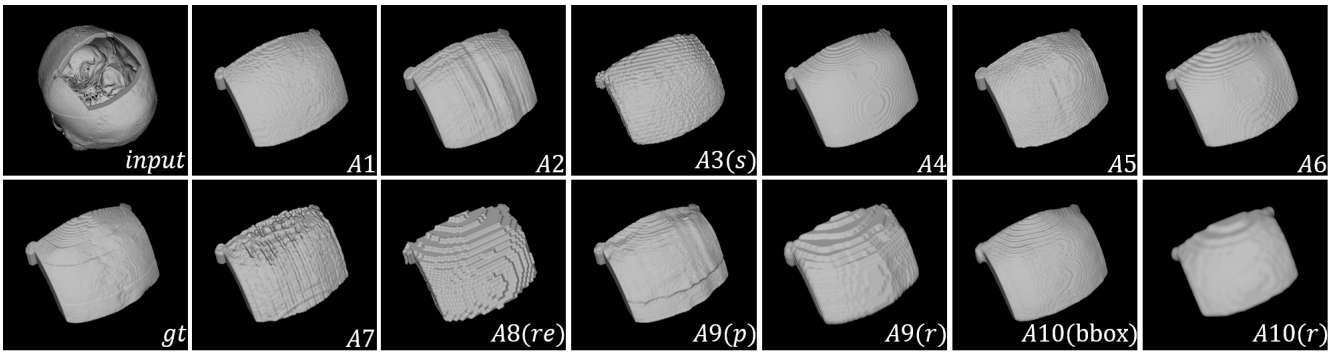


Fig. 8: Illustration of the implants (dimension $512^2 \times Z$) predicted by the algorithms. Left: the input defective skull and the corresponding ground truth implant. Right: the predictions for the algorithms.

shape completion. During training, the output of the RDU-Net and the corresponding ground truth is encoded into the latent feature space using the encoder part of the pre-trained VAE, and their distance in the latent space is used as a constraint in the learning process, which forces the network to produces anatomically and geometrically plausible skulls. Applying the shape constraint during the training process is similar to the shape fitting stage of the SSM method, where the prior knowledge about the shape of complete skulls is exploited on-the-fly. Besides, this course of action also diverts the attention of the network from the defects to the shape of the skull and thus eases the overfitting to defect patterns. The qualitative and quantitative comparison of $A8$ and $A8$ (re), according to Figure 6 (b) and Figure 7, shows the advantages of using the shape constraints. However, the t-test reported in Table II reveals that the improvement is not statistically significant regarding the quantitative metrics.

F. Post-Processing

Post-processing refers to the final steps taken in order to refine the output, including noise removal, hole filling, etc. These steps are closely related to the choice of problem formulation illustrated in Figure 3.

If the implant is obtained by subtracting the defective skull from a reconstructed complete skull, the resulting implant tends to contain both noise at the implant boundaries and isolated noise, which comes from the mismatch between the two skulls outside of the defective area. Morphological opening can be used to remove the noise attached to the implant boundaries. The isolated noise can be removed by keeping only the largest component, i.e., the implant, identified via connected component analysis (CCA). $A1$ and $A4$ applied morphological opening and CCA to the implant sequentially. For $A6$, after selecting the largest component using CCA, a spherical topological filter [54] was used to remove the attached noise non-destructively; a morphological closing and anti-aliasing filter was used to fill holes interior of the implant.

Direct implant prediction leads to isolated and attached noise as well. Unlike implants obtained from subtraction, directy predicted implants suffer mainly from attached noise. As before, isolated noise can be removed using CCA, and morphological opening can be used to remove attached noise [44]. However, morphological opening tends to remove not only noise but also fine details of the implant. Thus, $A5$ used a detail-preserving strategy to suppress such over-smoothing. An additional morphological dilation operation is applied to the implant after opening, which makes the implant slightly larger than the original implant. A clean implant preserving the fine details can be obtained by masking the original implant with the dilated implant using a logical *AND* operation.

G. Skull Dimension

The high dimensionality of the skull data posed a major challenge, as it was required that the predicted implants should be of the same dimension as the corresponding skulls. Direct processing of high-dimensional skulls is, in many situations, not feasible due to hardware limitations (see the *Hardware* column in Table III). Therefore, most of the algorithms down-sampled the skulls to a smaller size before submitting them into the network, as can be seen from the *Input dim* column in Table III. However, downsampling can cause loss of image quality, and learning from low-quality images yields coarse output (e.g., Figure 8, $A8$ (re)). Comparing these predictions with the ground truth, we can see that the surface of the implants produced by $A8$ (re), $A9$ (r) and $A10$ (r) is severely degraded with terracing artifacts, which is undesirable for this task. These algorithms used standard image resizing (interpolation) techniques to upsample the output to the original dimension for submission and cannot restore surface details.

To produce both high-dimensional and high-quality implants, three different strategies were explored: (1) $A3$, $A4$ and $A6$ reduce the image size before downsampling, as already discussed in Section IV (B) and Table IV. (2) $A5$ and $A9$ use patch-based training. (3) $A5$, $A7$, and $A10$ (bbox) apply a coarse-to-fine framework.

1) Patch-based Training: Dividing an image volume into several smaller patches and using these patches to train the network is a commonly applied strategy to deal with high-dimensional data [22]. Using a patch-based training method can lead to substantial improvement of the implant quality, as demonstrated by the comparison of $A9$ (r) and $A9$ (p) in Figure 8. We can see that there are obvious terracing artifacts on the surface of the implant from $A9$ (r) while the implant surface of $A9$ (p) is much smoother. Quantitatively, Figure 7

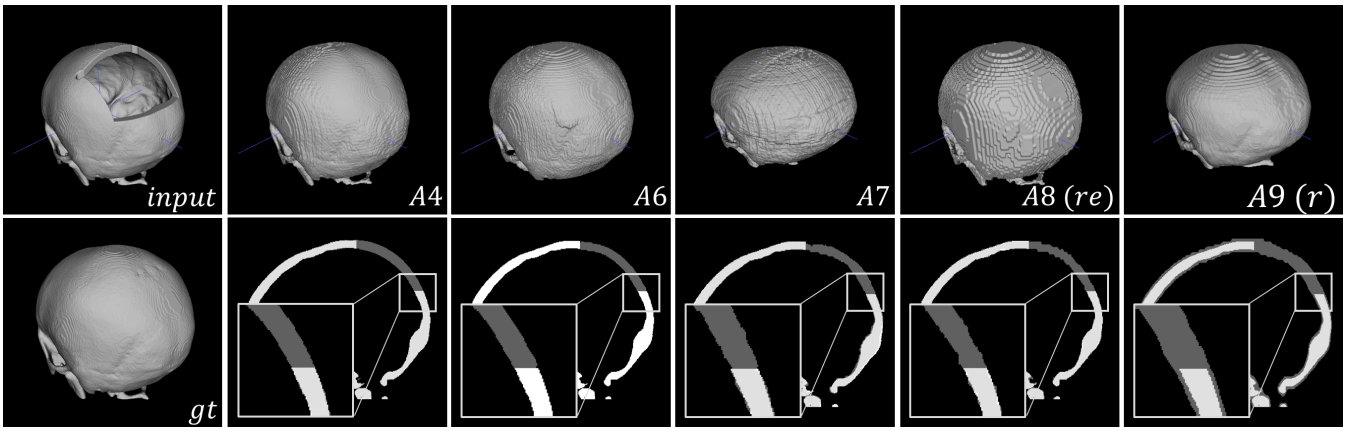


Fig. 9: An illustration of the reconstructed skulls by $A4$, $A6$, $A7$, $A8$ (re) and $A9$ (r), given as input a defected skull shown on the top left. The second row shows the ground truth skull in 3D and an overlay of the reconstructed skull (dark gray) onto the defective skull (light gray) in sagittal view.

and Table II also show that $A9$ (p) outperforms $A9$ (r) in terms of DSC and HD by a large margin (statistically significant).

2) Coarse-to-fine Framework: $A5$, $A7$ and $A10$ (bbox) adopted a *coarse-to-fine* framework to produce the desirable implants in two steps; each step is based on a deep neural network. The initial network is trained on downsampled skulls and therefore produces coarse implants. The second network produces fine implants based on the initial coarse prediction. For $A5$ and $A7$, upsampling a coarse implant, while at the same time restoring the geometric details on the implant surface is cast as a volumetric *super-resolution* task. For $A10$ (bbox), the coarse implant from the first network is used to extract the defective region ($256^2 \times 128$) on the original high-dimensional skull, and the second network predicts the fine implant directly from the extracted region, which is much smaller than the original volume. Figure 8 shows a comparison of the implants produced by $A10$ (r), which used standard interpolation for upsampling, and $A10$ (bbox). We can see that the implant from $A10$ (r) looks coarse and blurred on the surface while the implant from $A10$ (bbox) is of much higher quality. $A10$ (bbox) also beats $A10$ (r) regarding DSC and HD according to Figure 7. For DSC, the improvement of $A10$ (bbox) over $A10$ (r) is statistically significant according to Table II. Despite $A10$ (bbox) having better performance than $A10$ (r), its model is significantly more lightweight than that of $A10$ (r) as can be seen in Table III (# Param).

V. Discussions

A. Desired Algorithms Characteristics

From both a technical and application perspective, good generalization performance for various cranial defects and the ability to produce high-resolution and high-quality implants with affordable hardware (e.g., a desktop GPU) are among the most desirable characteristics of the algorithms for this challenge. For deep learning methods, the use of shape priors and defect augmentation can effectively increase the robustness. Besides, a statistical shape model (SSM) of the skull, which represents the general shape of a skull population and is independent from the defects, theoretically has the best generalization ability in this regard. However, the disadvantage of SSM methods is that inference tends to take much longer than with deep learning methods, up to 7-12 minutes per case [38]. The robustness of both deep learning and SSM to highly-deformed skulls is restricted to the training samples and can only be increased effectively by including representative deformed cases in the training phase. Processing high-dimensional 3D data, such as the skull data in this challenge, requires ample memory, often exceeding the capacity of commodity hardware. Downsampling the data as a workaround results in severe degradation of image quality. A two-step *coarse-to-fine* strategy, as used by $A5$, $A7$ and $A10$ (bbox) proves to be a solution to this problem.

For the algorithm produced implants, another desired characteristic is that the implants should be in congruency with the skulls in terms of shape and boundary for cosmetic and functional considerations. Figure 9 shows in 3D the reconstructed skulls by $A4$, $A6$, $A7$, $A8$ (re) and $A9$ (r), given as input a defective skull shown on the top left. It shows that these algorithms can successfully complete the defective skull and restore the missing skull bone, while the surface quality of the reconstructed skulls differs, similar to the implants shown in Figure 8. The reconstructed skulls are further overlaid onto the defective skull to examine how well they overlap in 2D sagittal views. On the defected region shows the difference between the reconstructed and defective skulls, i.e., the implant that can be obtained via a subtraction process illustrated in Figure 3. Ideally, a reconstructed skull should have a 100% overlap with the defective skull except on the defected region, and the implant should fit the skull in terms of shape (e.g., the surface curvature) and bone thickness on the edges. Figure 10 shows an implant created by the winning algorithm ($A4$) overlaid onto the corresponding defective skull. From the 3D view, we can see that the shape of the implant is compatible with its surrounding skull structures in terms of shape and boundary, so that the skull aesthetics can be restored. From the 2D views, we can see that the implant fits tightly against the defect edges on both the interior and exterior skull surfaces. We consider the tight edge contact a desirable characteristic for the implants

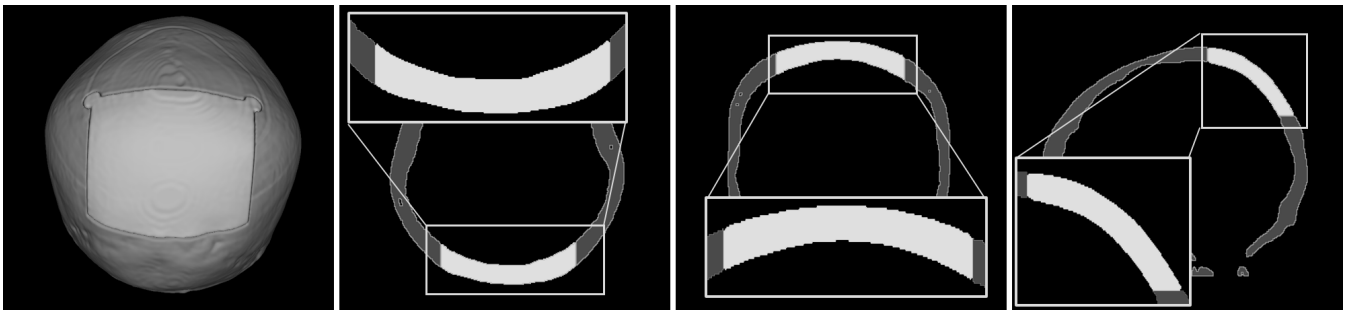


Fig. 10: An overlay of the implant from the winning algorithm ($A4$) onto a defective skull, viewed in 3D, axial, sagittal and coronal plane. The implanted area is zoomed in for the 2D views. To differentiate, skull and implant are in different colors.

produced by participants' algorithms [4].

Extrapolating from the findings of our MICCAI challenge, we see the following directions worthy of future study:

- Expand the training population for SSM and deep learning models, curate collections of head CT (normal, pathological, pediatric etc.) as training datasets [48].
- Explore alternative ways to create and incorporate shape priors or constraints of the skull into deep learning.
- Develop tools to generate synthetic defects from healthy skulls so that they closely resemble the defects from craniotomy, craniectomy and trauma.
- Preprocess the skulls before training to increase the learning efficiency.
- Extend current deep learning methods to data structures other than a voxel grid representation, such as a point cloud or mesh.

B. Limitations of the 1st AutoImplant Challenge

1) Dataset: The synthetic defects provided for training and evaluation in AutoImplant 2020 challenge are realistic, but simplified compared to real clinical defects from craniotomy and trauma. Defective skulls are hard to obtain clinically, let alone make public to participants, due to both the rarity of such operations and privacy restrictions. Even if defective skulls from clinical routine could be provided for evaluation, there often lack ground truth (implants), making quantitative evaluations impractical. It is therefore hard to judge the performance of the submitted algorithms on real defects. However, even if only synthetic defects were used, this challenge tried to encourage participants to improve the generalization performance of their algorithms to varied skull defects through either defect augmentation or the incorporation of skull shape priors, which have been proven effective to obtain top rankings in the challenge. For this reason, two test sets, $D_{test100}$ and D_{test10}, were used during the evaluation stage. $A1$ [38] is one of the representative algorithms with excellent generalization performance regarding skull defects, as the algorithm is built only upon healthy skulls and thus is independent from defective pattern.

Another limitation is the small number of unique skulls provided for training. While participants could create indefinite synthetic defects per skull to enlarge the training set and increase the defect variations, shape variations of the skulls were limited to the original 100 skulls provided. No algorithms had data to generalize to pathologically deformed skulls. This limitation can only be overcome by including more skulls in the training set. Hence, we have devised and open-sourced a pipeline [48] to convert collections of head CT, which are much easier to acquire than clinical defective skulls, into trainable datasets for the purpose of cranial implant design. At the core of the pipeline lies the creation of defective skulls out of complete skulls through the injection of synthetic defects. The pipeline can be extended to inject more realistic defects, or even allow multiple defects at once. Such a pipeline can also encourage the incorporation of skull data from different scanners, protocols, or populations into training.

2) Evaluation Metrics: Two quantitative metrics, DSC and HD, were used for the evaluation and ranking of the algorithms. The predicted implant that matches exactly with the ground truth (highest DSC and lowest HD) fits precisely with the defective area on the skull. However, instead of fitting exactly with the defect on the defect boundary, clinically usable implants should be minimally fault-tolerant in case of bone growth (ossification), the presence of scar tissues and osteolysis at the edge of the defects. Furthermore, cranial implant design is an *ill-posed* problem: An infinite variety of implants can serve the purpose of restoring the mechanical, protective and aesthetic functions. In other words, the ground truth used in the challenge is just one of the many possible solutions. However, current quantitative metrics constrain the solution to the ground truth, and other implants that are equally clinical usable, are penalized during scoring. More work will be needed to formalize the subjective judgement of neurosurgeons based on their professional experience. Besides, the implant boundaries are considered to be critical in cranioplasty and therefore should be given more emphasis compared to other parts of the implant during the evaluation phase. Current metrics, however, treat the implant as a whole and the boundary areas are not distinguished. Boundary-specific evaluation metrics are therefore highly desired in a future edition of the challenge.

[4]Note that, in cranioplasty, as the cranial implant is made of non-elastic materials. If necessary, neurosurgeons need to manually rasp the implant borders to enable the insertion of the implant onto the patient's skull.

C. Commercially Designed **Versus** Algorithm Produced Implants

In this section, we discuss how far the implants generated by the participants' algorithms are to the commercially designed implants, which are currently the clinical standard. According to our collaborating neurosurgeons, the actual cranial implants used in cranioplasty are usually thinner than the skull bone on the defected region, so that the interior surface of the implants will not apply pressure to the brain (more precisely, to the dura mater). Besides, a clinically usable implant does not necessarily have a tight contact with the skull on the edges. Instead, smalls gaps (in the order of one millimeter) around the borders of the implant and the skull defect are allowed and sometimes preferred, taking into consideration the bone regeneration over time. Therefore, when necessary, even the commercially designed and manufactured implants require some manual post-processing (e.g., rasping) by neurosurgeons before they can be used, especially when the design and manufacturing of the cranial implant takes a long time [5].

However, our challenge was designed to generate implants that can tightly fit the skull defects, as can be seen from Figure 9 and Figure 10. By doing so, the implants produced by the participants' algorithms can be further post-processed and rasped where necessary. Conversely, a too small implant is neither usable nor remediable via post-processing.

VI. Conclusions

This paper is aimed at giving a comprehensive overview of the first AutoImplant challenge hosted at MICCAI 2020. Contributions, approaches, evaluation results and algorithmic trends have been presented and discussed. We also included a critical judgement of current limitations for practical usage from clinical partners. With numerous participants and contributions from academia and industry around the world, the challenge provided a strong stimulus for automatic cranial implant design. To date, the challenge website remains open for post-challenge registrations and submissions, which has been accepted by the community as demonstrated by dozens of new registrations since the official end of the first challenge deadline. Should there be a future edition of the AutoImplant challenge, real defective skulls from craniotomy should be provided and the neurosurgeons' judgement on the clinical usability of the predicted implants should be involved in the evaluation phase. The scope could also be expanded to other medical scenarios involving computer-aided implant design, such as the lower jawbone [55] or ribs [56].

Acknowledgment

Jianning Li, Antonio Pepe, Christina Gsaxner, Yuan Jin, Matthias Eder, Dieter Schmalstieg and Jan Egger are with the Institute of Computer Graphics and Vision, Graz University of Technology, Inffeldgasse 16, 8010 Graz, Austria (e-mail: jianning.li@icg.tugraz.at).

Ulrike Zefferer, Gord von Campe, Karin Pistracher and Ute schäfer are with the Department of Neurosurgery, Medical University of Graz, Auenbruggerplatz 29, Graz, Austria (e-mail: gord.von-campe@medunigraz.at).

Victor Alves is with the Center Algoritmi, University of Minho, Braga, Portugal (e-mail: valves@di.uminho.pt).

Pedro Pimentel, Angelika Szengel, Moritz Ehlke, Hans Lamecker, Stefan Zachow and Heiko Ramm are with 1000shapes GmbH, Berlin, Germany (e-mail: info@1000shapes.com).

Stefan Zachow is with Zuse Institute Berlin (ZIB), Germany.

Laura Estacio is with San Pablo Catholic University, Arequipa, Peru.

Christian Doenitz is with the Department of Neurosurgery, University Medical Center Regensburg, Regensburg, Germany.

Haochen Shi is with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China (e-mail: jcsyshc@sjtu.edu.cn).

Xiaojun Chen is with the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China (e-mail: xiaojunchen@sjtu.edu.cn)

Franco Matzkin and Enzo Ferrante are with the Research Institute for Signals, Systems and Computational Intelligence, sinc(i), CONICET, FICH-UNL, Santa Fe, Argentina (e-mail: fmatzkin@sinc.unl.edu.ar).

Virginia Newcombe is with the University Division of Anaesthesia, University of Cambridge, Box 93, Addenbrooke's Hospital, Hills Road, Cambridge CB2 0QQ, UK.

Ben Glocker is with the Biomedical Image Analysis Group, Department of Computing, Imperial College London, UK.

David G. Ellis and Michele R. Aizenberg are with the Department of Neurosurgery, University of Nebraska Medical Center, Omaha, NE 68198 USA (e-mail: david.ellis@unmc.edu).

Oldřich Kodym, Michal Španěl and Adam Herout are with the Department of Computer Graphics and Multimedia, Brno University of Technology, Božetěchova 2, 612 66 Brno, Czech Republic (e-mail: ikodym@fit.vut.cz).

James G. Mainprize, Zachary Fishman, and Michael R. Hardisty are with the Sunnybrook Research Institute, 2075 Bayview Ave., Toronto, ON, Canada (e-mail: james.mainprize@sri.utoronto.ca).

James G. Mainprize is also with the Calavera Surgical Design Inc., Toronto, ON, Canada. and Michael R. Hardisty is also with the Division of Orthopaedic Surgery, University of Toronto, Toronto, ON, Canada (e-mail: m.hardisty@utoronto.ca).

Amirhossein Bayat, Suprosanna Shit and Bjoern H. Menze are with the Department of Informatics, Technical University of Munich, Boltzmannstr. 3, Garching bei München, 85748, Germany (e-mail: amir.bayat@tum.de).

Bjoern H. Menze is with the Department for Quantitative Biomedicine, University of Zurich, Switzerland (e-mail: bjoern.menze@tum.de).

Bomin Wang and Zhi Liu are with the School of Information Science and Engineering, Shandong University, Qingdao, China (e-mail: 201712354@mail.sdu.edu.cn).

References

[1] R. Stefini, G. Esposito, B. Zanotti, C. Iaccarino, M. M. Fontanella, and F. Servadei, "Use of "custom made" porous hydroxyapatite implants for cranioplasty: postoperative analysis of complications in 1549 patients," *Surgical neurology international*, vol. 4, 2013.

[2] A. Morais, "Automated computer-aided design of cranial implants-a deep learning approach," Master's thesis, Universidade do Minho, 2018.

[3] J. V. Rosenfeld and J. W. Tee, "Complications after decompressive craniectomy and cranioplasty," in *Complications in Neurosurgery*, pp. 266–273, Elsevier, 2019.

[4] D. B. Kurland, A. Khaladj-Ghom, J. A. Stokum, B. Carusillo, J. K. Karimy, V. Gerzanich, J. Sahuquillo, and J. M. Simard, "Complications associated with decompressive craniectomy: a systematic review," *Neurocritical care*, vol. 23, no. 2, pp. 292–304, 2015.

[5] G. von Campe and K. Pistracher, "Patient specific implants (psi): Cranioplasty in the neurosurgical clinical routine," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 1–9, Springer, 2020.

[6] J. Li, A. Pepe, C. Gsaxner, and J. Egger, "An online platform for automatic skull defect restoration and cranial implant design," in *Medical Imaging 2021: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 11598, p. 115981Q, International Society for Optics and Photonics, 2021.

[7] X. Chen, L. Xu, X. Li, and J. Egger, "Computer-aided implant design for the restoration of cranial defects," *Scientific Reports*, pp. 1–10, 2017.

[8] A. Marzola *et al.*, "A semi-automatic hybrid approach for defective skulls reconstruction," *Computer-Aided Design and Applications*, vol. 17, pp. 190–204, 2019.

[9] M. Gall, X. Li, X. Chen, D. Schmalstieg, and J. Egger, "Computer-aided planning and reconstruction of cranial 3d implants," *IEEE Engineering in Medicine and Biology Society*, pp. 1179–1183, 2016.

[10] J. Egger *et al.*, "Interactive reconstructions of cranial 3D implants under MeVisLab as an alternative to commercial planning software," *PLoS ONE*, vol. 12, p. 20, 2017.

[11] L. Mei, M. Figl, A. Darzi, D. Rueckert, and P. Edwards, "Sample sufficiency and PCA dimension for statistical shape models," in *European Conference on Computer Vision*, pp. 492–503, Springer, 2008.

[12] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014.

[13] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.

[15] J. Carr, W. R. Fright, and R. Beatson, "Surface interpolation with radial basis functions for medical imaging," *IEEE Transactions on Medical Imaging*, vol. 16, pp. 96–107, 1997.

[16] W. Semper-Hogg *et al.*, "Virtual reconstruction of midface defects using statistical shape models.," *Journal of cranio-maxillo-facial surgery*, vol. 45(4), pp. 461–466, 2017.

[17] M. A. Fuessinger *et al.*, "Virtual reconstruction of bilateral midfacial defects by using statistical shape modeling," *Journal of Craniomaxillofacial Surgery*, vol. 47, pp. 1054–1059, 2019.

[18] M. A. Fuessinger *et al.*, "Planning of skull reconstruction based on a statistical shape model combined with geometric morphometrics," *International Journal of Computer Assisted Radiology and Surgery*, vol. 13, pp. 519–529, 2017.

[19] H. Lamecker, *Variational and statistical shape modeling for 3D geometry reconstruction*. PhD thesis, Zuse-Institut Berlin, 2008.

[20] Z. Kun, "Dense correspondence and statistical shape reconstruction of fractured, incomplete skulls," Master's thesis, National University of Singapore, 2014.

[21] A. Morais, J. Egger, and V. Alves, "Automated computer-aided design of cranial implants using a deep volumetric convolutional denoising autoencoder," in *World Conference on Information Systems and Technologies*, pp. 151–160, Springer, 2019.

[22] J. Li, "Deep learning for cranial defect reconstruction," Master's thesis, Graz University of Technology, 1 2020.

[23] O. Kodym, M. Španěl, and A. Herout, "Skull shape reconstruction using cascaded convolutional networks," *Computers in Biology and Medicine*, vol. 123, p. 103886, 2020.

[24] F. Matzkin, V. Newcombe, S. Stevenson, A. Khetani, T. Newman, R. Digby, A. Stevens, B. Glocker, and E. Ferrante, "Self-supervised skull reconstruction in brain CT images with decompressive craniectomy," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 390–399, Springer, 2020.

[25] Y. Zhang, Y. Pei, Y. Guo, S. Chen, T. Xu, and H. Zha, "Cleft volume estimation and maxilla completion using cascaded deep neural networks," in *International Workshop on Machine Learning in Medical Imaging*, 2020.

[26] X. Han, Z. Li, H. Huang, E. Kalogerakis, and Y. Yu, "High-resolution shape completion using deep neural networks for global structure and local geometry inference," in *IEEE International Conference on Computer Vision*, pp. 85–93, 2017.

[27] A. Dai, C. Ruizhongtai Qi, and M. Nießner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5868–5877, 2017.

[28] X. Wen, T. Li, Z. Han, and Y.-S. Liu, "Point cloud completion by skip-attention network with hierarchical folding," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1939–1948, 2020.

[29] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "Pcn: Point completion network," in *International Conference on 3D Vision*, pp. 728–737, IEEE, 2018.

[30] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "Shapenet: An information-rich 3D model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[31] M. Liu, L. Sheng, S. Yang, J. Shao, and S.-M. Hu, "Morphing and sampling network for dense point cloud completion," in *AAAI Conference on Artificial Intelligence*, vol. 34, pp. 11596–11603, 2020.

[32] P. Liepa, "Filling holes in meshes," in *Eurographics Symposium on Geometry Processing*, pp. 200–205, 2003.

[33] V. Kraevoy and A. Sheffer, "Template-based mesh completion," in *Eurographics Symposium on Geometry Processing*, vol. 385, pp. 13–22, 2005.

[34] A. Prutsch, A. Pepe, and J. Egger, "Design and development of a web-based tool for inpainting of dissected aortae in angiography images," *arXiv preprint arXiv:2005.02760*, 2020.

[35] K. Armanious, V. Kumar, S. Abdulatif, T. Hepp, S. Gatidis, and B. Yang, "ipA-MedGAN: Inpainting of arbitrary regions in medical imaging," in *IEEE International Conference on Image Processing*, pp. 3005–3009, IEEE, 2020.

[36] N. Gapon, V. Voronin, R. Sizyakin, D. Bakaev, and A. Skorikova, "Medical image inpainting using multi-scale patches and neural networks concepts," in *IOP Conference Series: Materials Science and Engineering*, vol. 680, p. 012040, IOP Publishing, 2019.

[37] J. V. Manjón, J. E. Romero, R. Vivo-Hernando, G. Rubio, F. Aparici, M. de la Iglesia-Vaya, T. Tourdias, and P. Coupé, "Blind MRI brain lesion inpainting using deep learning," in *International Workshop on Simulation and Synthesis in Medical Imaging*, pp. 41–49, Springer, 2020.

[38] P. Pimentel *et al.*, "Automated virtual reconstruction of large skull defects using statistical shape models and generative adversarial networks," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 16–27, Springer, 2020.

[39] H. Shi and X. Chen, "Cranial implant design through multiaxial slice inpainting using deep learning," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 28–36, Springer, 2020.

[40] F. Matzkin, V. Newcombe, B. Glocker, and E. Ferrante, "Cranial implant design via virtual craniectomy with shape priors," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 37–46, Springer, 2020.

[41] D. G. Ellis and M. R. Aizenberg, "Deep learning using augmentation via registration: 1st place solution to the AutoImplant 2020 challenge," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 47–55, Springer, 2020.

[42] O. Kodym, M. Španěl, and A. Herout, "Cranial defect reconstruction using cascaded CNN with alignment," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 56–64, Springer, 2020.

[43] J. G. Mainprize, Z. Fishman, and M. R. Hardisty, "Shape completion by U-Net: An approach to the AutoImplant MICCAI cranial implant design challenge," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 65–76, Springer, 2020.

[44] A. Bayat, S. Shit, A. Kilian, J. T. Liechtenstein, J. S. Kirschke, and B. H. Menze, "Cranial implant prediction using low-resolution 3D shape completion and high-resolution 2D refinement," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 77–84, Springer, 2020.

[45] B. Wang *et al.*, "Cranial implant design using a deep learning method with anatomical regularization," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 85–93, Springer, 2020.

[46] Y. Jin, J. Li, and J. Egger, "High-resolution cranial implant prediction via patch-wise training," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 94–103, Springer, 2020.

[47] J. Li, A. Pepe, C. Gsaxner, G. von Campe, and J. Egger, "A baseline approach for autoimplant: the MICCAI 2020 cranial implant design challenge," in *Multimodal Learning for Clinical Decision Support and Clinical Image-Based Procedures*, pp. 75–84, Springer, 2020.

[48] J. Li and J. Egger, "Dataset descriptor for the autoimplant cranial implant design challenge," in *Towards the Automatization of Cranial Implant Design in Cranioplasty*, pp. 10–15, Springer, 2020.
[49] J. Li and J. Egger, *Towards the Automatization of Cranial Implant Design in Cranioplasty: First Challenge, AutoImplant 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 8, 2020, Proceedings*. Lecture Notes in Computer Science, Springer International Publishing, 2020.
[50] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.
[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
[52] P. Shamsolmoali, M. Zareapoor, R. Wang, H. Zhou, and J. Yang, "A novel deep structure U-Net for sea-land segmentation in remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 9, pp. 3219–3232, 2019.
[53] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *International Conference on 3D Vision*, pp. 565–571, IEEE, 2016.
[54] M. Styner *et al.*, "Framework for the statistical shape analysis of brain structures using SPHARM-PDM," *Insight Journal*, no. 1071, 2006.
[55] L. Nickels, "World's first patient-specific jaw implant," *Metal Powder Report*, vol. 67, no. 2, pp. 12–14, 2012.
[56] J. Kang, L. Wang, C. Yang, L. Wang, C. Yi, J. He, and D. Li, "Custom design and biomechanical analysis of 3d-printed peek rib prostheses," *Biomechanics and modeling in mechanobiology*, vol. 17, no. 4, pp. 1083–1092, 2018.