Hole-filling framework by combining structural and textural information for the 3D Terracotta Warriors

Tong Chu, Wenmin Yao, Jie Liu, Xueli Xu, Haiyang Nan, Xin Cao,* Kang Li, and Mingquan Zhou

Northwest University, School of Information and Technology, National-Local Joint Engineering Research Center of Cultural Heritage Digitization, Xi'an, China

Abstract. As one of the eight wonders in the world, the virtual restoration of Terracotta Warriors is of great significance to archaeology. However, some parts of fragments were corroded for thousands of years, resulting in the existence of several holes in most of the restored cultural heritage artifacts. Based on the structural and textural information, we present a framework for filling the hole. First, a method based on the Poisson equation was employed to fill the hole on the triangular mesh model. Then, to complete the surface color and texture information of the three-dimension (3D) model, make the hole patch, and the original model surface texture natural transition, the 3D problem is converted into two-dimension (2D) image inpaint problem, and a refined network is added into EdgeConnect to generate a higher resolution result. A set of experiments is performed to evaluate the performance of our proposed framework. We hope the proposed framework can provide a useful tool to guide the virtual restoration of other cultural heritage artifacts. © *2021 Society of Photo-Optical Instrumentation Engineers (SPIE)* [DOI: 10.1117/1.JRS.15.046503]

Keywords: hole completion; Terracotta Warriors; neural network; point cloud; cultural relic restoration.

Paper 210529 received Aug. 19, 2021; accepted for publication Oct. 28, 2021; published online Nov. 12, 2021.

1 Introduction

The Terracotta Warriors and Horses of Qin Shihuang are the political, military, and cultural symbols of the ancient Qin Dynasty in China and are the precious cultural heritage of human beings.[1–4] There are three pits in the Terracotta Army of the Qin Dynasty and thousands of life-size Terracotta Warriors with different facial features. Besides, there are weapons, ceramic horses, and chariots [Fig. 1(a)]. Due to time-induced natural corrosion and other reasons, many Terracotta Warriors were in a damaged state when unearthed. To reproduce the form of cultural relics, the matching and restoration of cultural relics have become one of the main tasks of archaeologists. With the development of computer vision technology, computer-assisted virtual restoration technology (CAVR) has become one of the powerful tools to convert cultural relics into 3D models and guide the restoration of cultural relics. The main step in CAVR is the reassembly of the broken fragments. The related strategies generally fall into two categories: geometry-driven methods (also called the automatic methods) and semiautomatic methods. The geometry-driven methods are based on the completeness and smoothness of the fracture-regions, whereas the semiautomatic methods incorporate expert opinions and knowledge to select the fragments that can be assembled. However, as the fracture surfaces of many fragments are eroded, the matched fragments cannot be glued tightly, resulting in the existence of holes on the assembled relics [Fig. 1(b)].

Filling these holes on the surface of cultural relics reasonably is of great significance for the subsequent development of virtual restoration technology of cultural relics.[5–9] In the virtual display, the surface color and texture information of 3D cultural relics are also particularly important. Therefore, to address the problem of missing surface texture after structural repair of 3D

*Address all correspondence to Xin Cao, xin_cao@163.com

Fig. 1 (a) The image of the pit K9901 of the Terracotta Army; and (b) the assembled 3D model of the Terracotta Warrior.

mesh. Based on the traditional geometric repair algorithm and deep neural network, we proposed the 3D cultural relic hole repair method, which can repair texture information while filling the hole 3D information and produce a textured high-resolution model. Besides, for the 2D image inpainting, we added a refine network after the original EdgeConnect to construct the improved network, which makes the transition of texture information of the restored cultural relic surface more natural. Finally, the textural image is projected onto the 3D mesh model by utilizing the Mudbox software to obtain the completed model.

To summarize, the main contributions are as follows:

(1) A hole filling framework for the Terracotta Warriors is proposed, which may inspire the restoration of other cultural relics in the future.
(2) The framework can repair the texture information while filling holes, and the final complete model is with high-resolution textural information.
(3) Experiments demonstrate that our framework can produce higher quality repair results compared with the existing methods.

The remainder of the paper is organized as follows. In Sec. 2, the related works are described. The methods, network architecture, and our framework are described in Sec. 3. The experimental results and analysis are provided in Sec. 4. Discussion and conclusion are presented in Sec. 5.

2 Related Work

2.1 *3D Shape Completion*

Many hole-filling algorithms have been proposed in recent years. In general, existing algorithms can be categorized into two classes: traditional geometric-based methods and deep learning-based methods. On the one hand, the traditional geometric methods repair the holes based on template matching and mesh surface fitting. Deep learning methods are divided into voxel-based and point cloud-based completion methods according to the different representations of 3D objects.

Patched-based methods are suitable for repairing the larger holes. The easiest way is to find patches similar to the missing area from the 3D template library to fill the holes.[10,11] Chaudhuri and Koltun[12] and Kim et al.[13] retrieved the closest shape with the shape database. Li et al.[14] searched objects from the 3D data shape database and obtained a model that closely matched

the original combination through alignment and scaling. Guo-Hua et al.[15] selected the best matching template to fill the incomplete model by constructing feature matching candidate sets. Sahay and Rajagopalan[16] created a depth map database of the 3D model, projected the missing model into a 2D depth map, matched the best depth map in the template library, and restored it to a 3D partial surface to fill the hole. However, due to the limited data of the template library, the repair cannot be completed when the incomplete model does not match the template library.

Therefore, many research methods use the nonlocal similarity of the 3D model to complete the hole repair.[17–19] Sharf et al.[20] and Park et al.[21] found the best matching patch of the hole according to the geometric information of the given model and transplanted similar regions into the hole. Fu et al.[22] divided the point cloud into cubes of the same size, used the similarity measure to select the cube that is most similar to the missing area from the source area, and combined the graphic signal processing technology to fill the hole. Dinesh et al.[23] mainly selected adaptive templates based on geometric methods using the principle of nonlocal similarity and filled holes with block matching by calculating similarity. The above method can fill larger holes well, but at least one complete model similar to the model with holes is required.

The method based on mesh surface can be used to repair smaller holes and use the topological relationship of the 3D mesh to fit and reconstruct the missing region. Fortes et al.[24] proposed a method to use the reconstruction function based on the 3D grid to combine the effective region features to fit and reconstruct the missing regions. Lin et al.[25] proposed a surface fitting reconstruction method based on tensor voting for discrete point cloud surfaces. Combining the advantages of the edge projection-based 3D reconstruction method and the radical basis function (SFM) method, Gai et al.[26] proposed a hole filling fitting algorithm based on SFM. The projection method is used to extract the hole boundary from the 2D depth map, and the radial basis function is used to complete the hole information, which can effectively restore the information of the complex surface hole. Besides, many methods directly act on the missing region. According to the effective features around the hole, the fitting plane is obtained by interpolation within the hole.[27,28] Zhang and Zhou[29] constructed a symbolic distance function, used the variational level set method to extract holes, and repaired the entire missing area by iterative diffusion based on effective voxels. Ngo and Lee[30] used the curve around the hole to reconstruct the characteristic curve of the missing area, divided the hole into several smaller sub-holes, and then filled the subholes. However, because traditional geometric methods use information such as curvature and topological structure to reconstruct, they cannot accurately repair sharp and complex geometric features, and it takes a long time to run the algorithm and cannot handle larger and more holes on 3D models. Li et al.[31] proposed a method that performs better even holes with more sharp curvature. This work fitted and reconstructed the predicted surface based on the Poisson equation and adjusted the normal vector of the missing region to stitch the patch and the original model to complete the hole repair.

Many learning-based methods are developed to inpainting regular holes. Earlier work used a voxel convolutional network to complete the incomplete model,[32–37] but some geometric information will be lost when the model is voxelized, which limits the model resolution.

With the proposal of PointNet,[38] it is possible for the convolutional network to directly act on the disordered point cloud. Many studies use the encoder to convert input point cloud data into latent vectors and then use the decoder to decode back to point cloud data.[39–41] Yang et al.[42] raised a method based on the combination of autoencoder and GAN and reconstructed high-dimensional accurate 3D structures from a single depth map repair the object. Chen et al.[43] combined the autoencoder and the GAN network, took the potential features encoded by the autoencoder as the input of the generator, and trained the discriminator to recognize the point cloud model and the original model generated by the generator. Compared with traditional methods, deep learning-based methods have more reasonable results. However, the voxel-based model requires high computer memory, and due to the limitation of the input point cloud data, the generated point cloud model cannot guarantee the resolution. The purpose of this paper is to predict the surface texture information based on the completed patched and evenly integrate the effective missing area uniformly. So we use hole-filling based on Poisson equation,[31] which can obtain a uniform density with an effective area and restore the repaired mesh with original characteristics.

2.2 *2D Image Inpainting*

Broadly speaking, conventional image inpainting methods are divided into partial differential equation-based algorithms and sample patch-based algorithms. The former is mainly aimed at small-scale damage, using a single-pixel as the basic unit of predict, and using the principle of diffusion to complete information, but its robustness is poor and it is easy to cause blur. The latter finds the patch with maximum similarity from the known area and copies it to the defect area.

Recently, deep learning-based methods usually encode an image into a latent feature and decode the feature back into an image. Context encoder[44] combined the encoder–decoder and the GAN to predict the large-scale image by context pixel prediction and can generate a reasonable result. But the above method does not take into account the overall consistency of the image. Iizuka et al.[45] added global and local context discriminator to context encoder enable it to judge the generation effect from both global and local perspectives and can predict any irregularly shaped missing area. Yu et al.[46] copied and improved the latest recent repair model[45] to build its basic generated image repair network and introduce a coarse to refinement network structure, where the first network made initial coarse predictions, and the second network took the prediction as input and predicted higher-quality inpainting results. The existing convolution methods have shortcomings, because convolution layers act on all pixels, causing color inconsistency and blurriness. Partial convolution[47] was proposed to predict irregular holes, where the convolution is only performed in the valid area of the image (the mask part is 0), and the mask of the picture will continue to iterate and shrink as the number of layers of the network deepens. Existing deep learning approaches can generate coherent structures in the missing regions. But for the earth-colored 3D shape of the Terracotta Warriors, the image can be repaired according to the change of its surface texture. So the method based on edge prediction[48] is more suitable, which predict image by predicting the edge of the missing area, and our experiments show that it can produce superior results.

3 Methodology

3.1 *Hole Identification*

An incomplete 3D Terracotta Warriors model may have many different types of holes in different locations (e.g., Fig. 2). The hole can be classified into four types (erosion, reassembly, human,

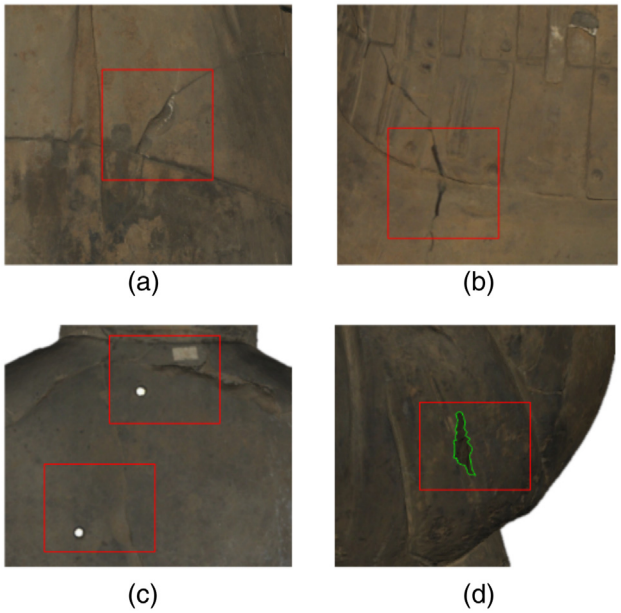

Fig. 2 The types of holes existed on the Terracotta Warriors: (a) erosion factor, (b) reassembly factor, (c) human factor, and (d) scanning factor.

Algorithm 1 Hole identification algorithm.

Initialize. An incomplete 3D Terracotta Warriors mesh model.

Input: Vertices V with vertex v_i, and a flag b_i whether the current vertex has been checked (the default is false)

Step 1: Select one of vertex v_i that b_i is false.

Step 2: Check every triangle m_i that contain v_i, and then loop every adjacent point v_j except v_i on the mesh m_i.

Step 3: If the b_j is false and $v_i v_j$ is the boundary line (the edge connects exactly one triangle mesh, because an edge of the triangle mesh can share two triangle meshes).

Step 4: The v_i and v_j are added into the hole edges set H.

Step 5: Repeat the step 2–4 until a closed hole is formed.

Step 6: Find other vertex and b_i is false.

Output. The vertices collection of holes.

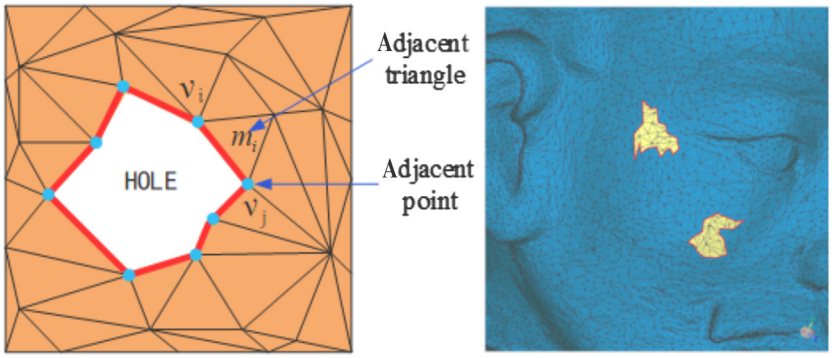

Fig. 3 (a) Schematic diagram of related concepts of holes and (b) examples of hole identification.

and scanning factors). Except for the scanning factor, the former three need to manually delete the original incomplete triangular meshes at the repaired location before identifying the hole. Due to the lack of topology information, it is challenging to accurately detect holes before repairing the model. Here, we propose a hole identification method as described in Algorithm 1.

In this paper, hole detection is applied on the 3D mesh model, which is composed of multiple triangular grid mosaics. As shown in Fig. 3(a), adjacent triangles are connected to the same triangular mesh vertices (e.g., m_i), and two vertices of an edge in the same triangle mesh are adjacent point (e. g., v_i and v_j are adjacent points to each other). According to the triangle principle, the one-adjacent triangle of the vertex of the complete 3D mesh model is equal to the number of one-adjacent points.

Therefore, we traverse the unvisited vertices first, to determine whether the number of triangles containing the vertex and the number of adjacent points of the vertex are equal, then to determine whether the vertex belongs to the hole vertices collection, until the hole vertices collection forms a closed loop [Fig. 3(b)]. Thinking of the situation contains more than a hole, we find the vertices are not visited for again until all vertices have been accessed.

3.2 *Structural Repair for Holes*

In this paper, a structural repair based on Poisson reconstruction is adopted.[31] The algorithm establishes the Poisson equation according to the normal vector of the 3D directed point set

and the Laplacian operator and then obtains the 3D surface equation by solving the Poisson equation and finally apply the marching cubes algorithm extracts the iso-surface of the equation to obtain the 3D surface, where the iso-surface extraction is to obtain points on the same surface. Poisson equation fits and reconstructs the input incomplete mesh model from the global and local perspectives, and this method can obtain smooth surfaces and avoid self-intersection. After the algorithm obtains the predicted surface, the direction of the generated triangular patch will be adjusted according to the normal vector information of the grid around the hole, so the generated patch can conform to the original feature distribution as much as possible. The flowchart is shown in Fig. 4.

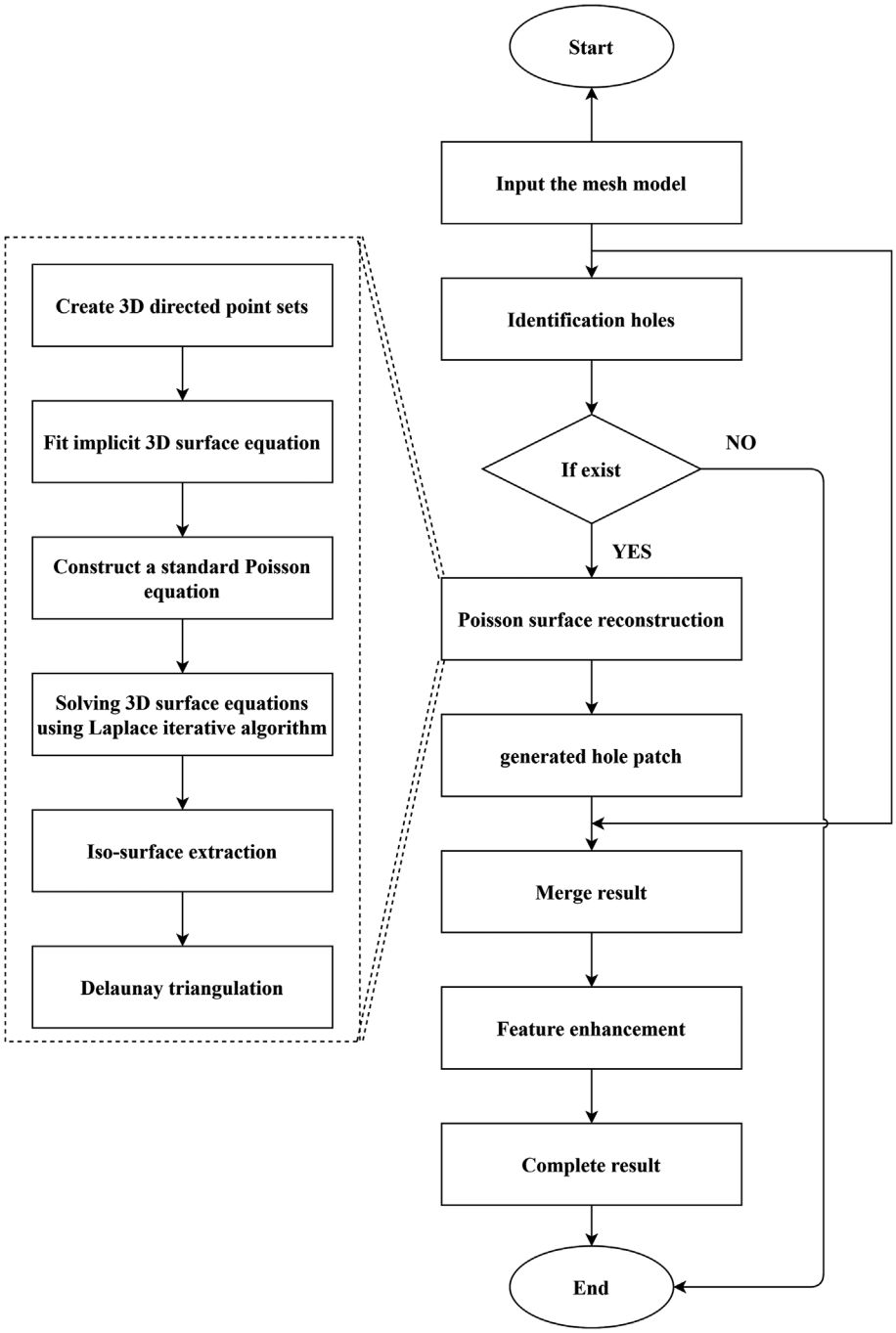

Fig. 4 The flowchart of the proposed structural repair method.

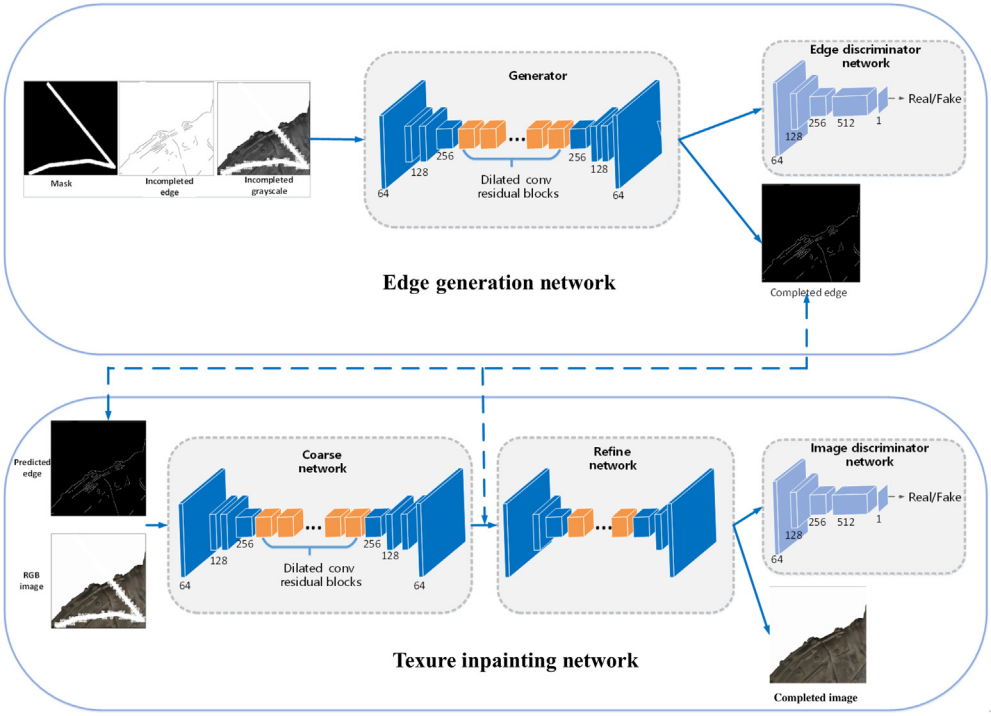

Fig. 5 The framework of texture repair method based on edge information.

3.3 *Texture Repair for Holes*

To repair the 3D hole topology while preserving the color and texture information of the surface of the Terracotta Warriors, an improved deep neural network for image inpainting is proposed. Inspired by the EdgeConnect,[48] a refined network is added in the original EdgeConnect second stages, the proposed network can generate the details of the Terracotta Warriors surfaces. The network can be divided into two parts: the edge generation network (EGN) and the texture inpainting network (TIN). The EGN produces the edge map for the masked region, whereas TIN uses the predicted edge to guide the repair of the missing texture of the image. Specifically, the TIN consists two modules: a coarse network and a refine network. There are also two discriminator networks: edge discriminator network and image discriminator network, which are used to predict an edge/inpainted image is real or not, and the discriminator uses the PatchGAN discriminator in Ref. 49. The architecture of the network is shown in Fig. 5.

3.3.1 *Edge generation network*

The P_{gt} and E_{gt} are the ground truth image and edge, P_{gray} is the grayscale image of P_{gt}. The input of EGN consists of three parts: image mask, the incompleted edge, and the incompleted grayscale image. Let $\overline{P_{gray}} = P_{gray} \odot (1 - M)$ and $E_{in} = E_{gt} \odot (1 - M)$ be the incompleted grayscale image and edge image. The mask image M as the precondition (1 for missing areas and 0 for background). The prediction output of the EGN is predicted edge image, which can be expressed as

$$E_{out} = G_{EGN}(M, E_{in}, \overline{P_{gray}}). \tag{1}$$

EGN is trained with an objective comprised of an adversarial loss and feature-matching loss as follows:

$$\ell_{EGN} = \alpha \cdot (\min_{G_{EGN}} \ell_{G_{EGN}} + \min_{D_{EDN}} \ell_{D_{EDN}}) + \beta \cdot \min_{G_{EGN}} \ell_{FM}, \tag{2}$$

where α and β are the regularization parameters, ℓ_{FM} is the feature-matching loss.[50] ℓ_{EGN} is the composed loss of EGN, which is defined as the least square loss function [Eq. (3) and (4)]. The $\ell_{D_{\mathrm{EDN}}}$ and $\ell_{G_{\mathrm{EGN}}}$ are the loss of discriminator and generator, respectively. It can generate higher image accuracy and make the network training process more stable, which are different with EdgeConnect:

$$\min_{G_{\mathrm{EGN}}} \ell_{G_{\mathrm{EGN}}} = \frac{1}{2} E_{(P_{\mathrm{gray}}, E_{\mathrm{out}})}[D(E_{\mathrm{out}}) - 1]^2, \tag{3}$$

$$\min_{D_{\mathrm{EGN}}} \ell_{D_{\mathrm{EGN}}} = \frac{1}{2} E_{(P_{\mathrm{gray}}, E_{\mathrm{gt}})}[D(E_{\mathrm{gt}}) - 1]^2 + \frac{1}{2} E_{(P_{\mathrm{gray}}, E_{\mathrm{out}})}[D(E_{\mathrm{out}})]^2. \tag{4}$$

3.3.2 *Texture inpainting network*

The TIN has two modules: the coarse (TIN^1) and refine (TIN^2) networks. We input the predicted edge E_{out} and the incomplete RGB image $P_{\mathrm{rgb}} = P_{\mathrm{gt}} \odot (1 - M)$ into the coarse network, which outputs a coarse RGB image. To infer a more accurate result, the refine network is adopted, which takes the coarse image P_{coarse} and the former predicted edge E_{out} as input. Finally, a reality and more precise image is generated. And the refine network has a similar architecture as the coarse network. The prediction output of the TIN can be expressed as

$$P_{\mathrm{coarse}} = G^1_{\mathrm{TIN}}(E_{\mathrm{out}}, P_{\mathrm{rgb}}), \tag{5}$$

$$P_{\mathrm{refine}} = G^2_{\mathrm{TIN}}(E_{\mathrm{out}}, P_{\mathrm{coarse}}), \tag{6}$$

where G^1_{TIN} and G^2_{TIN} are the coarse network and refine network of the TIN module, respectively. The loss of TIN is a combination of ℓ_1 loss, adversarial loss ℓ_{TIN}, perceptual loss[51] ℓ_{perc}, and style loss[52] ℓ_{sty}. Specifically, the TIN module includes two generators and one discriminator. The final loss can be defined as

$$\ell_{\mathrm{TIN}} = a \cdot (\ell^1_{G_{\mathrm{TIN}}} + \ell^2_{G_{\mathrm{TIN}}} + \ell^2_{D_{\mathrm{TIN}}}) + b \cdot (\ell^1_{\mathrm{prec}} + \ell^2_{\mathrm{prec}}) + c \cdot (\ell^1_{\mathrm{sty}} + \ell^2_{\mathrm{sty}}) + d \cdot (\ell^1_1 + \ell^2_1), \tag{7}$$

where the adversarial loss is defined similar to EGN, as follows:

$$\min_{G_{\mathrm{TIN}}} \ell^1_{G_{\mathrm{TIN}}} = \frac{1}{2} E_{(P_{\mathrm{coarse}}, E_{\mathrm{out}})}[D(P_{\mathrm{coarse}}) - 1]^2, \tag{8}$$

$$\min_{G_{\mathrm{TIN}}} \ell^2_{G_{\mathrm{TIN}}} = \frac{1}{2} E_{(P_{\mathrm{refine}}, E_{\mathrm{out}})}[D(P_{\mathrm{refine}}) - 1]^2, \tag{9}$$

$$\min_{D_{\mathrm{IDN}}} \ell_{D_{\mathrm{IDN}}} = \frac{1}{2} E_{(P_{\mathrm{gt}}, E_{\mathrm{out}})}[D(P_{\mathrm{gt}}) - 1]^2 + \frac{1}{2} E_{(P_{\mathrm{refine}}, E_{\mathrm{out}})}[D(P_{\mathrm{refine}})]^2. \tag{10}$$

More details can refer to EdgeConnect,[48] and in this article, the above-mentioned parameters are set as $\alpha = 1$, $\beta = 10$, $a = b = 1$, $c = 100$, $d = 1$.

After the repaired texture image is generated, we map the texture image to the surface of the 3D mesh model. The work is done by the graphic editing software Autodesk Mudbox.

3.4 *Proposed Framework*

Based on the above methods, the flowchart of our proposed hole filling framework for the Terracotta Warriors is shown in Fig. 6. The proposed framework can repair the texture information of the surface while completing the 3D structure. Since the whole Terracotta Warriors model contains a large number of triangle mesh. And it occupies a large memory, which affects the efficiency of the algorithms. Therefore, we only extract the patches with holes as the input of

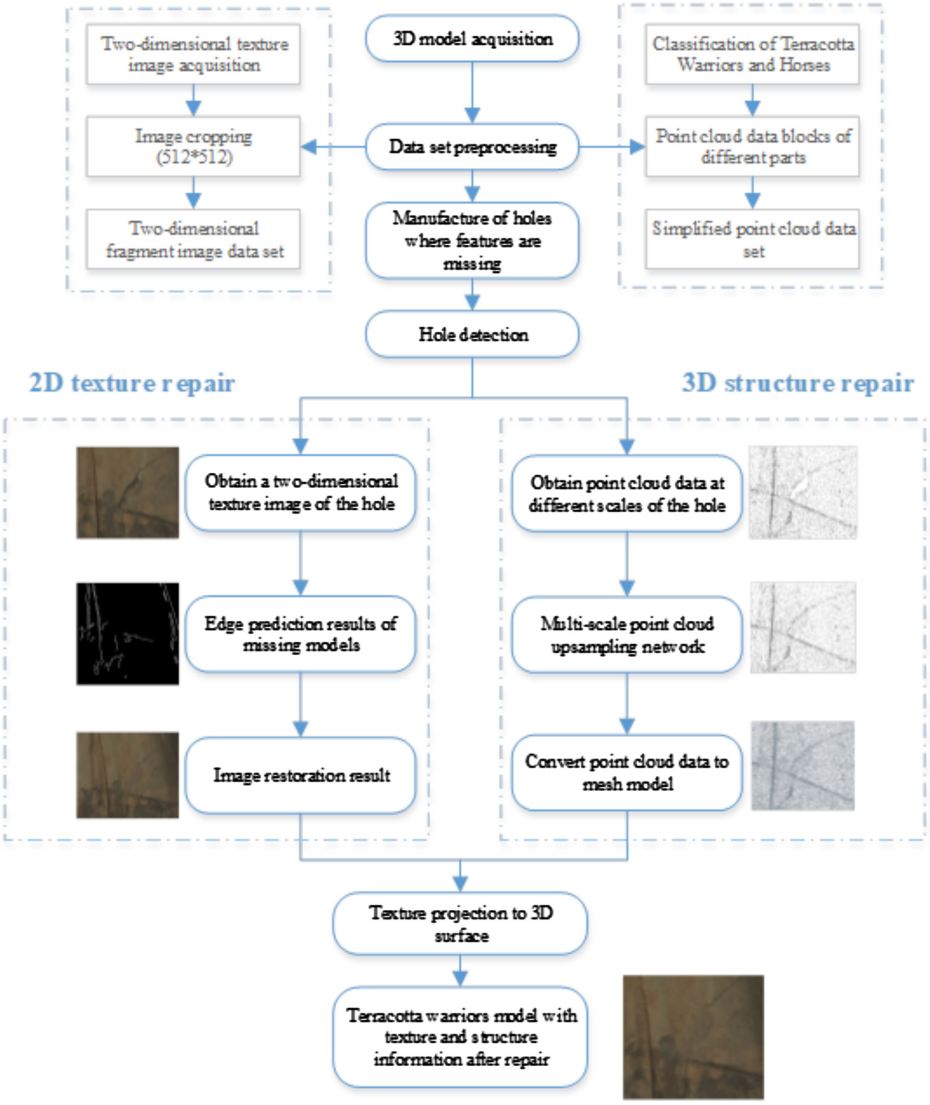

Fig. 6 The overall framework.

the algorithm and to improve the execution efficiency of the algorithm. It is worth noting that during the experiment, enough effective areas will be guaranteed to achieve the real complement effect. Besides, the denoise process will be done before our experience by the Arctect when combining the raw scan data.

First, we apply Algorithm 1 to detect holes in the virtual mesh model. After locating the position of the holes, the images and mesh patches containing the hole are obtained. Second, the feature enhancement algorithm based on Poisson's equation is used to complete the mesh structure, and the network based on edge prediction is applied to repair the image. Third, the repaired 2D surface texture information is mapped to the 3D mesh surface by Mudbox. Finally, the 3D model patch with structure information and surface textual information is merged with the initial model to obtain a complete repair result.

4 Experiment and Results

4.1 *Dataset and Implementation Details*

In this paper, the fragments of the Terracotta Warriors unearthed in pit K9901 of Qin Shihuang's mausoleum are used. The point cloud of these fragments was scanned using a Creaform VIU

handy scanner. The scan resolution was 3.91 mm, which favors speed but results in relatively low precision. The point cloud exhibits a strong local imbalance in the sampling pattern and contains realistic noise that was the result of the scanning process. To assemble these fragments, a multi-feature fusion method proposed in Ref. 53 is used. The 3D structural repair is an algorithm that does not require prelearning, therefore the denoised incomplete model can directly input and output the completed result. By performing the coordinate conversion, rotation, and normalization operations on the 3D model with color texture, the image dataset for training is obtained in batches.

The 3D structural repair is implemented with the Visual Studio 2019 and OpenGL, whereas the texture repair network is implemented by Pytorch. The hardware of the PC is i7-7820 CPU/3.6 GHz, 64GiB memory, and RTX TITAN GPU.

4.2 *Results of Structural Completion*

Figure 7 shows the results of hole completion in the 3D mesh based on Poisson's equation. The algorithm can generate uniform and reasonable mesh patches for different parts of the Terracotta Warriors models. It can be seen from Fig. 7(b) that the generated patch boundary cannot perfectly match the input model, resulting in a partial depression at the boundary. But the merging result is smoother after feature enhancement [see Fig. 7(c)]. For completing a hole with larger curvatures, such as the nose, the completed triangular mesh is dented after the initial repair, and repaired patch can be more consistent with the original model after feature enhancement. As shown in Fig. 7, the algorithm can produce the same density as the original mesh and less self-intersection even the 3D mesh with complex features and can improve the calculation efficiency.

4.3 *Results of Texture Completion*

Because the effective area around the hole has a great effect on filling the missing content, and the surface information far away from the hole plays a small role in repairing the hole. Considering computer memory, it is necessary to preprocess the original Terracotta Warriors model for training. We crop the 2D image to generate a 256*256 image patch. A total of 25,933 image patches extracted from 80 Terracotta Warriors models are used.

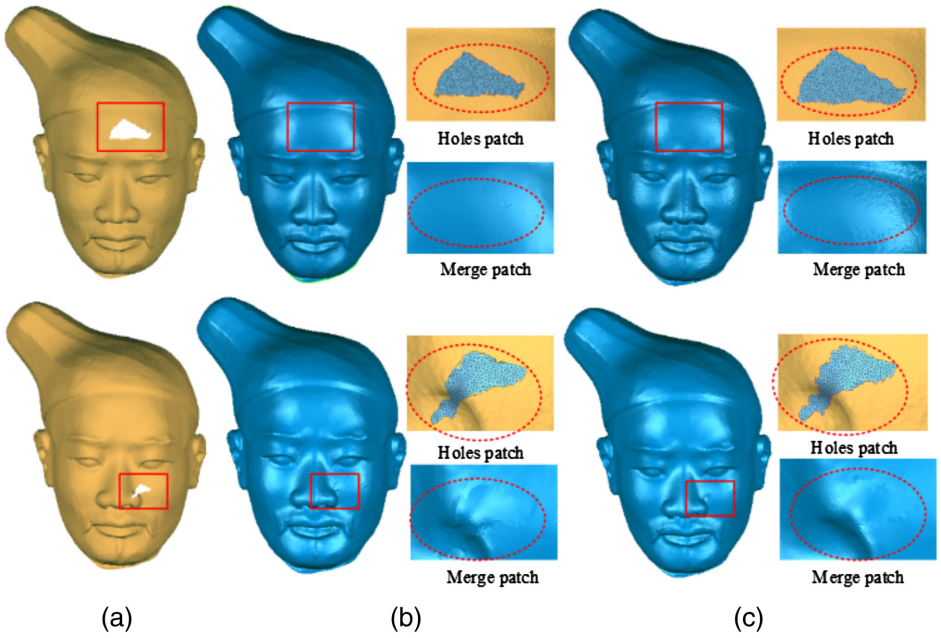

Fig. 7 The result of structural repair of the head of the Terracotta Warriors. N-FM indicates the process without feature enhancement, and FM indicates feature enhancement of the triangular mesh: (a) input, (b) output (N-FM), and (c) output (FM).

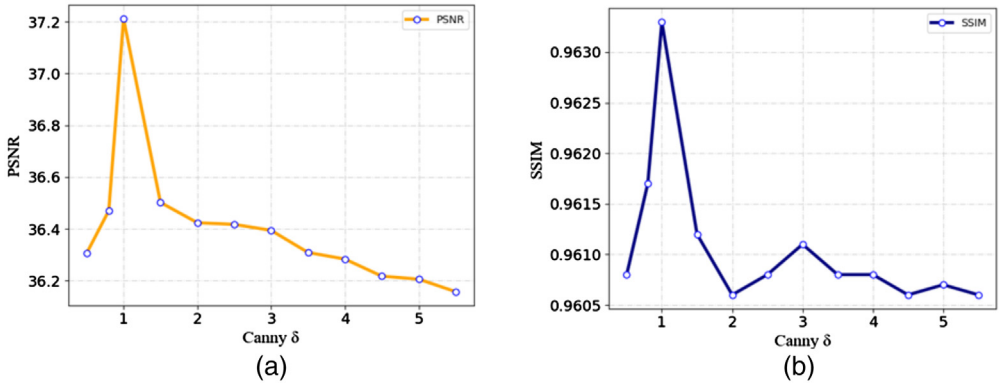

Fig. 8 The impact of Gaussian smoothing filter δ of Canny edge detector on the texture repair results: (a) PSNR and (b) SSIM[55] are used.

The irregular masks are obtained from the work of Liu et al.[47] For training, the missing image is generated by combining the mask with the original image, and the edge maps are generated using the Canny edge detector. It is worth noting that the training dataset contains scenes with different views for various parts of the model. During the training, the batch size is set to 8, and the total epochs of 1,000,000 are set to ensure that the network can converge completely. The optimization operator is Adam[54] with $\beta_1 = 0$ and $\beta_2 = 0.9$. The EGN and TIN are trained separately with a learning rate of 10^{-4} until the losses plateau, and then lower the learning rate to 10^{-5} and continue to train EGN and TIN until convergence.

To verify the impact of the Gaussian smoothing filter δ of Canny edge detector on the texture repair results, some preliminary experiments are conducted. From Fig. 8, it is obvious that δ should be set 1 to obtain the best experimental effect. When δ is too large, there are too few edges to guide the texture restoration correctly. In contrast, there are too many edges when δ is too small, and it can mislead the generated image.

Figure 9 shows inpainting results under the change of δ. The first one in the first line is the image to be repaired with mask, and others are the predicted edge images. The second line only shows the region of completed holes. And the first one in the second line is ground truth, the others are the repair results obtained based on the predicted edge. The best result can be generated when $\delta = 1$. Although the mask in the figure is small, it can be seen that when δ is greater than 2, the fusion of the generated repair patch and the original image is reduced, and the repair quality is degraded.

We use the Terracotta Warriors dataset to fine-tune based on the parameters of the first and second phases of EdgeConnect, and in the third phase of the refined network, we use the parameters obtained in the second phase to fine-tune. When generating the valid set, we randomly generate single or multiple holes for each 2D image of Terracotta Warriors. To verify the effectiveness of the improved method is compared with the state-of-the-art image inpainting methods on the Terracotta Warriors Dataset. The improved methods include two deep learning-based

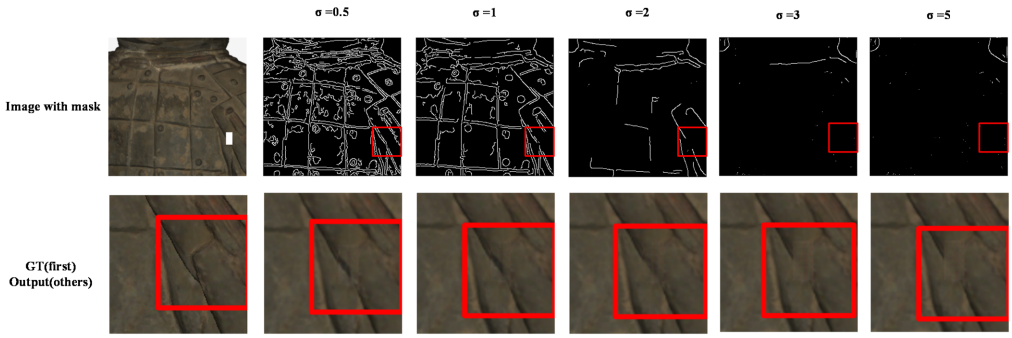

Fig. 9 The effect of the edges generated by different δ values on the inpainting image.

Table 1 Quantitative results of different methods.

Methods	PatchMatch	Global&Local	EdgeConnect	Ours
PSNR	36.8368	35.6802	37.0322	**37.2125**
SSIM	0.9584	0.9533	0.9640	**0.9668**
MAE	0.0050	0.0051	0.0046	**0.0043**

models: EdgeConnect[48] and Global&Local,[45] and the traditional inpainting method PatchMatched.[56]

During the test, we used common evaluation metrics, i.e., peak signal-to-noise ratio (PSNR), structural similarity (SSIM),[55] and mean square error (MAE), which is calculated by the complete image and the ground-truth image in pixel space, to quantify the performance of the models. Table 1 shows the evaluation results, the higher the values of PSNR and SIMM, the better result of network predict, whereas the MAE is the opposite. In the deep learning-based method, our model outperforms the existing methods on all three metrics. From the results, we can see that the PatchMatch method is better than Global&Local. The results can be explained that the surface color is earthy and the model has the nature of local similarity, so the method based on patched matching has better repair than the deep learning method. Furthermore, Edgeconnect has a better prediction should be due to the edge-based guidance. When the refine network was added, the model improved performance on all three evaluation indicators, the PSNR, SSIM, and MAE value is increased by 0.486%, 0.290%, and 6.97%, respectively, compared with EdgeConnect, which shows the effectiveness of the improved network.

Figure 10 shows a visual comparison of our method with existing methods. As can be seen from the figure, PatchMatch[56] generates a smooth texture. But for images with obvious feature changes, the result is not consistent in color. Global&Local[45] has a large difference in the color of the generated results, which cannot be well integrated with the original model. EdgeConnect[48] can generate smooth and reasonable images because of boundary guidance and the addition of an attention model, but there are still subtle differences. After adding the refine network, the object has a very natural boundary and a coherent surface with the neighborhood context. In addition, the image with the surface depression due to corrosion can be masked to make the repaired surface more realistic and coherent (e.g., the last line).

4.4 *Hole Repair Results*

Three groups of experiments are conducted to verify the proposed framework. The holes in the first group are caused by natural causes, as shown in Fig. 11. Most of the holes existed in the fragment splicing areas [see Fig. 11(a)]. The mask, which denotes the area to be repaired, is first generated manually. It should be slightly larger than the original hole region, as shown in Fig. 11(d). The structural repair results are shown in Fig. 11(e), and the textual repair results are shown in Fig. 11(f). The final repair results are shown in Fig. 11(b).

The second holes are caused by human factors. The holes in Fig. 12(a) are caused by artificial marks. The repair result is shown in Fig. 12(b), and more details can be found in Figs. 12(c)–12(f), which demonstrate that proposed framework can deal with these holes well.

Figure 13 shows the holes caused by natural reasons or scanning reasons, which contains many irregular missing holes on the surface [see Fig. 13(c)]. The repair result is shown in Fig. 13(b) demonstrates the proposed framework has good repair results for repairing the missing areas of complex physical structures and textures.

4.5 *Texture Mapping*

In this paper, Mudbox is used to map the 2D texture image back to the surface of the repaired model of the 3D Terracotta Warrior structure. By corresponding to spatial vertices, the information such as color and texture is projected onto the surface, and then automatically deforming

Fig. 10 Qualitative comparison with the state-of-the-art method. From left to the right are: input image with holes, PatchMatch,[56] Global&Local,[45] EdgeConnect,[48] our improvement model, and the ground-truth, respectively.

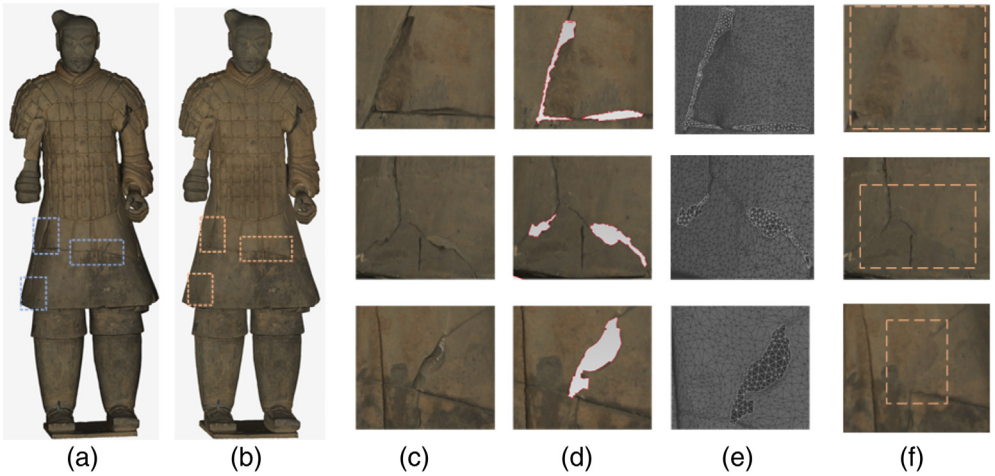

Fig. 11 Repair results of the holes that were caused by natural causes: (a) the original model, (b) the repair results, (c) details of the hole area, (d) masks of the area to be repaired, (e) structure repair results, and (f) texture repair results.

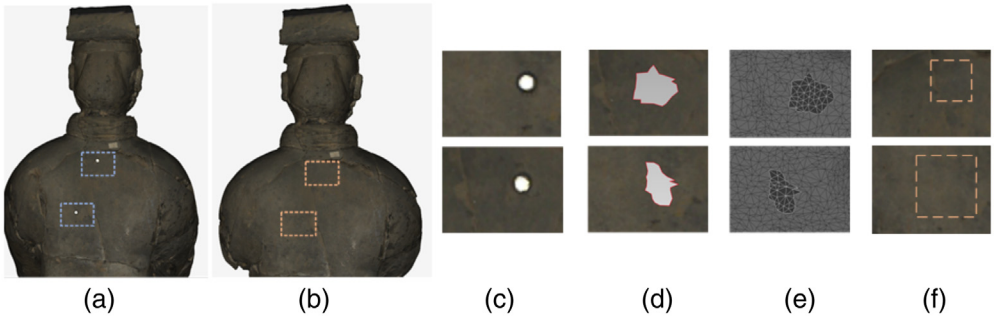

Fig. 12 Repair result of the holes caused by human factors: (a) The original model, (b) the repair result, (c) details of the hole area, (d) mask of the area to be repaired, (e) structure repair results, and (f) texture repair results.

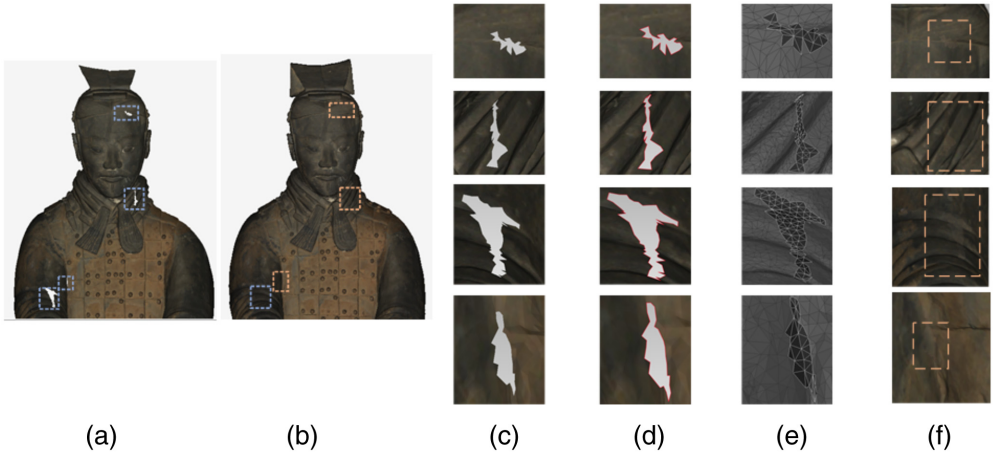

Fig. 13 Repair result of the holes caused by natural reasons and scanning reasons: (a) the original model, (b) the repair result, (c) details of the hole area, (d) mask of the area to be repaired, (e) structure repair result, and (f) texture repair result.

Fig. 14 Comparison of the model before and after repair. (a) Before repair and (b) after 3D repair, the result of mapping the 2D image to the 3D surface.

the model based on its depth, to make it more suitable for 3D surface curvature changes to cover the surface of the 3D Terracotta Warriors. The feature master module of Mudbox is utilized to map the feature, 2D image information is assigned to the 3D model using tools, such as moving, rotating, adjusting the brush radius, the JPG file of the repaired UV line is exported, and then the UV file of the original model is replaced. We finally obtain the 3D Terracotta Warrior model after the structure and texture restoration are obtained (Fig. 14).

Table 2 Quantitative comparison of different loss functions used in the TIN stage.

Loss	PSNR	SSIM	MAE
a	36.8368	0.9633	0.0051
B	36.8927	0.9646	0.0049
C	37.0322	0.9654	0.0045
D	**37.2125**	**0.9668**	**0.0043**

4.6 *Other Experiments*

To prove the necessity of loss function, we studied the loss function of TIN stage. (a) The second stage has only the least squares GANs; (b) the loss function includes the ℓ_{GAN} and ℓ_1 loss functions; (c) adds the perceptive loss function ℓ_{perc} on top of the b loss function; (d) adds the joint loss function after the style loss ℓ_{sty}; and contains EPN, TIN^1, and TIN^2. As shown in Table 2, the network model repair effect gradually improves as the loss function increases. The reason is that the four loss functions have different functions, the generate adversarial loss function is to make the network generate more realistic repair results, whereas the loss function ℓ_1 is to judge the difference between the predict results and the missing image, and the perceived loss function ℓ_{perc} to ensure that the real image and the repair image are more similar, and the style loss function ℓ_{sty} to ensure that the repair results of the missing areas without feature loss.

5 Discussion and Conclusion

As an important carrier of cultural heritage, many cultural relics have been damaged to varying degrees due to the impact of environmental and human factors. The virtual restoration of cultural relics is the process of using the scanned 3D model to go through a series of process operations to obtain the original model after restoration. We propose a 3D cultural relics restoration framework, which can restore the 3D physical structure while retaining the surface texture information. Taking the Terracotta Warriors as an example, by identifying the holes in the 2D cultural relics, using the Poisson-based reconstruction method for surface reconstruction, and enhancing the characteristics of the reconstructed triangular patches, then a smooth surface can be obtained. In texture restoration, combining neural networks and GAN networks and leveraging edge information to guide the inpainting process, the coarse and refine two-stage network model can produce more coherent texture information, and finally generate more surface authenticity, texture coherence, and structural integrity relics. Experimental results show that the framework of this

Fig. 15 Inpainting results of different repair inputs patch with holes, where (a) and (b) represent two different 3D Terracotta Warriors model.

paper can produce good results in the field of 3D cultural relics with different types of multiple irregular shape missing areas.

In addition, to improve the efficiency of the algorithm, we use local patches as input. Figure 15 shows that different patch regions containing the same holes have different repair results. So when we extract the input, which should include as much effective information around the holes as possible. Because it is believed that the closer feature to the hole, the greater effect of repairing the hole. Of course, the framework of this paper also has certain limitations. For holes with large missing areas, by reason of the algorithm directly calculate the output results without prelearning, as a result, the quality of 3D structural repair is reduced. Therefore, in future work, we will study the use of deep learning technology to use a large number of datasets to learn the potential features of the Terracotta Warriors and horses to repair and improve the efficiency and accuracy of repair.

In general, this work proposes a 3D virtual heritage restoration framework. The restoration framework is established by combining 3D structure completion and 2D texture inpainting. Experimental results show that the framework has a good restoration effect. We hope that this work can promote the development of digital restoration of cultural relic models in the field of computer graphics.

Acknowledgments

We thank the Emperor Qinshihuang's Mausoleum Site Museum for providing the Terracotta Warriors data. This work was supported in part by the National Key Research and Development Program of China (Grant No. 2019YFC1521103); China Post-doctoral Science Foundation (Grant No. 2018M643719); Young Talent Support Program of the Shaanxi Association for Science and Technology (Grant No. 20190107); Key Research and Development Program of Shaanxi Province (Grant No. 2019GY-215); Major research and development project of Qinghai Grant No. 2020-SF-143).

References

1. P. S. Quinn et al., "Building the Terracotta Army: ceramic craft technology and organisation of production at Qin Shihuang's mausoleum complex," *Antiquity* **91**(358), 966–979 (2017).
2. W. Zilin, "The museum of Qin Shi Huang Terracotta Warriors and horses," *Museum Int.* **37**(3), 140–147 (1985).
3. J. Wang, Z. Hui, and L. Li, "Research on digitizing processing of the Terracotta Warriors and Horses of the Qin dynasty," in *Int. Conf. Geometric Model. and Graphics*, pp. 201–207 (2003).
4. H. Kinoshita, "The first emperor: China's Terracotta Army: exhibition at the British museum, 13 September 2007–6 April 2008," *Asian Aff.* **38**(3), 371–376 (2007).
5. M. Dellepiane et al., "Using digital 3D models for study and restoration of cultural heritage artifacts," in *Digital Imaging for Cultural Heritage Preservation: Analysis, Restoration, Reconstruction of Ancient Artworks*, pp. 39–70 (2011).
6. M. Pieraccini, G. Guidi, and C. Atzeni, "3D digitizing of cultural heritage," *J. Cult. Herit.* **2**(1), 63–70 (2001).
7. R. Scopigno , M. Callieri, and P. Cignoni et al., "3D models for cultural heritage: beyond plain visualization," *Computer* **44**(7), 48–55 (2011).
8. C. Serain, "The contribution of digital technologies to the mediation of the conservation-restoration of cultural heritage," in *Euro-Mediterranean Conf.*, pp. 283–289, Springer (2016).
9. F. Wang, "A study of digital image enhancement for cultural relic restoration," *Int. J. Eng. Tech. Res.* **11**(7), 41–44 (2017).
10. N. J. M. Mark Pauly et al., "Example-based 3D scan completion," in *Eurographics Symp. Geometry Process.*,H. P. M. Desbrun, Ed., pp. 1–10 (2005).
11. J. Rock et al., "Completing 3D object shape from one depth image," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 2484–2493 (2015).

12. S. Chaudhuri and V. Koltun, "Data-driven suggestions for creativity support in 3D modeling," *ACM Trans. Graphics* **29**(6), 1–10 (2010).
13. Y. M. Kim et al., "Guided real-time scanning of indoor objects," *Comput. Graphics Forum* **32**(7), 177–186 (2013).
14. Y. Li et al., "Database-assisted object retrieval for real-time 3D reconstruction," *Comput. Graphics Forum* **34**(2), 435–446 (2015).
15. G. Guo-Hua et al., "Three-dimensional model restoration with shape-constrained sample filling," *Opt. Precis. Eng.* **26**, 2863–2872 (2018).
16. P. Sahay and A. N. Rajagopalan, "Geometric inpainting of 3D structures," in *IEEE Conf. Comput. Vision and Pattern Recognit. Workshops (CVPRW)*, pp. 1–7 (2015).
17. C. Dinesh, I. V. Bajić, and G. Cheung, "Exemplar-based framework for 3D point cloud hole filling," in *IEEE Vis. Commun. and Image Process. (VCIP)*, pp. 1–4 (2017).
18. G. Harary, A. Tal, and E. Grinspun, "Context-based coherent surface completion," *ACM Trans. Graphics* **33**(1), 1–12 (2014).
19. P. Sahay and A. N. Rajagopalan, "Harnessing self-similarity for reconstruction of large missing regions in 3D Models," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR2012)*, pp. 101–104 (2012).
20. A. Sharf, M. Alexa, and D. Cohen-Or, "Context-based surface completion," in *ACM SIGGRAPH 2004 Papers*, Association for Computing Machinery, Los Angeles, California, pp. 878–887 (2004).
21. S. Park et al., "Shape and appearance repair for incomplete point surfaces," in *Tenth IEEE Int. Conf. Comput. Vision (ICCV'05) Volume 1*, Vol. **1262**, pp. 1260–1267 (2005).
22. Z. Fu, W. Hu, and Z. Guo, "Point cloud inpainting on graphs from non-local self-similarity," in *25th IEEE Int. Conf. Image Process. (ICIP)*, pp. 2137–2141 (2018).
23. C. Dinesh, I. V. Bajic, and G. Cheung, "Adaptive non-rigid inpainting of 3D point cloud geometry," *IEEE Signal Process. Lett.* **25**(6), 878–882 (2018).
24. M. A. Fortes et al., "Filling holes with geometric and volumetric constraints," *Comput. Math. Appl.* **74**(4), 671–683 (2017).
25. H. Lin et al., "Feature preserving filling of holes on point sampled surfaces based on tensor voting," *Math. Prob. Eng.* **2018**, 8076910 (2018).
26. S. Gai et al., "Research on a hole filling algorithm of a point cloud based on structure from motion," *J. Opt. Soc. Am. A* **36**(2), A39–A46 (2019).
27. V. Hoang et al., "Localization estimation based on Extended Kalman filter using multiple sensors," in *IECON 2013 - 39th Annu. Conf. IEEE Ind. Electron. Soc.*, pp. 5498–5503 (2013).
28. Y. Quinsat, "Filling holes in digitized point cloud using a morphing-based approach to preserve volume characteristics," *Int. J. Adv. Manuf. Technol.* **81**(1–4), 411–421 (2015).
29. J. Zhang and M. Zhou, "Repair methods of complex holes in a three-dimensional model based on variation level set," *Int. J. Pattern Recognit. Artif. Intell.* **32**(06), 1855007 (2018).
30. H. T. M. Ngo and W. S. Lee, "Feature-first hole filling strategy for 3D meshes," in *VISIGRAPP 2011. Commun. Comput. and Inf. Sci.*, pp. 53–68 (2013).
31. Y. Li, G. Geng, and X. Wei, "Hole-filling algorithm based on Poisson equation," **43**(10), 209–215, 221 (2017).
32. A. Dai, C. Qi, and M. Nießner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 6545–6554 (2017).
33. X. Han et al., "High-resolution shape completion using deep neural networks for global structure and local geometry inference," in *Int. Conf. Comput. Vision (ICCV)*, pp. 85–93 (2017).
34. A. Sharma, O. Grau, and M. Fritz, "VConv-DAE: deep volumetric shape learning without object labels," in *Comput. Vision – ECCV 2016 Workshops*, pp. 236–250 (2016).
35. D. Thanh Nguyen et al., "A field model for repairing 3D shapes," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 5676–5684 (2016).
36. J. Varley et al., "Shape completion enabled robotic grasping," in *IEEE/RSJ Int. Conf. Intell. Rob. and Syst. (IROS)*, pp. 2442–2447 (2017).

37. W. Wang et al., "Shape inpainting using 3D generative adversarial network and recurrent convolutional networks," in *IEEE Int. Conf. Comput. Vision (ICCV)*, pp. 2317–2325 (2017).
38. C. R. Qi et al., "Pointnet: deep learning on point sets for 3D classification and segmentation," in *Int. Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 652–660 (2017).
39. P. Achlioptas et al., "Learning representations and generative models for 3D point clouds," in *Int. Conf. Mach. Learn.*, pp. 40–49 (2018).
40. Y. Yang et al., "FoldingNet: point cloud auto-encoder via deep grid deformation," in *IEEE/CVF Conf. Comput. Vision and Pattern Recognit.*, pp. 206–215 (2018).
41. W. Yuan et al., "PCN: point completion network," in *Int. Conf. 3D Vision (3DV)*, pp. 728–737 (2018).
42. B. Yang et al., "3D object reconstruction from a single depth view with adversarial learning," in *IEEE Int. Conf. Comput. Vision Workshops (ICCVW)*, pp. 679–688 (2017).
43. X. Chen, B. Chen, and N. J. Mitra, "Unpaired point cloud completion on real scans using adversarial training," arXiv:00069 (2019).
44. D. Pathak et al., "Context encoders: feature learning by inpainting," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 2536–2544 (2016).
45. S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graphics* **36**(4), 1–14 (2017).
46. J. Yu et al., "Generative image inpainting with contextual attention," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 5505–5514 (2018).
47. G. Liu et al., "Image inpainting for irregular holes using partial convolutions," in *Proc. Eur. Conf. Comput. Vision*, pp. 85–100 (2018).
48. K. Nazeri et al., "EdgeConnect: generative image inpainting with adversarial edge learning," arXiv:00212 (2019).
49. P. Isola et al., "Image-to-image translation with conditional adversarial networks," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 5967–5976 (2017).
50. T.-C. Wang et al., "High-resolution image synthesis and semantic manipulation with conditional GANs," in *IEEE/CVF Conf. Comput. Vision and Pattern Recognit.*, pp. 8798–8807 (2018).
51. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Eur. Conf. Comput. Vision* (2016).
52. L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 2414–2423 (2016).
53. Y. Zhang et al., "A multi feature fusion method for reassembly of 3D cultural heritage artifacts," *J. Cult. Herit.* **33**, 191–200 (2018).
54. D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," arXiv:1412.6980 (2014).
55. Z. Wang, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**(4), 600–612 (2004).
56. M. Bleyer, C. Rhemann, and C. Rother, "PatchMatch stereo-stereo matching with slanted support Windows," *BMVC* **11**, 1–11 (2011).

Tong Chu is a postgraduate student at Northwest University. She received her BS degree in software engineering from Northwest University in 2017. Her current research interests include remote sensing and image processing.

Biographies of the other authors are not available.