

# AUTOMATIC TEXTURE MAPPING METHOD FOR 3D MODELS TO CIRCUMVENT OCCLUSION THROUGH 3D SPATIAL THROUGH-VIEW RELATIONSHIPS BETWEEN IMAGES, MODELS, AND POINT CLOUDS

Hao Liu<sup>1</sup>, Linfu Xie<sup>1</sup>, Xiaoming Li<sup>1,\*</sup>, W. X. Wang<sup>1</sup>

<sup>1</sup> Research Institute for Smart Cities, School of Architecture and Urban Planning, Shenzhen University, Shenzhen, PR China; 1900325007@email.szu.edu.cn; linfuxie@szu.edu.cn; lixmings@szu.edu.cn; wangwx@szu.edu.cn;

Commission II, WG II/4

**KEYWORDS:** Texture mapping, 3D reconstruction, Point clouds, Occlusion detection, MRF

## ABSTRACT:

Photorealistic 3D models play an important role in various applications related to smart cities. Texture mapping plays a crucial role as the last step of 3D reconstruction. The texture of the model directly reflects the visualization and realism of the model. In complex city scenes, objects often have varying degrees of occlusion from one another. Traditional methods to circumvent occlusion by image or model data are not very effective. In this paper, we propose an automatic texture mapping method for 3D models to circumvent occlusion through 3D spatial through-view relationships between images, models, and point clouds. First of all, calculate the density and voxel resolution of the point clouds, the point clouds here can be either dense matching point clouds or laser point clouds. Voxelization is based on point clouds, voxels generated by voxelization can reflect the spatial location of the point clouds. Secondly, to determine whether there is an occlusion of voxels between the image and the model vertices based on the spatial through-view relationship between the image, the model, and the voxels, and filter out the unobstructed images. Finally, by optimizing the data items of the Markov random field, the shading function is added to the original data items for evaluating the shading condition, based on which the most suitable image is selected for texture mapping of a triangular surface of the model. Loop computation to complete the unobstructed texture mapping of the entire model. In this paper, the public test datasets released by the International Society for Photogrammetry and Remote Sensing (ISPRS) and several urban area datasets collected by ourselves are incorporated to test the performance of the proposed method. Both qualitative and quantitative cross-sectional comparisons with existing texture mapping algorithms are conducted and presented. Experimental results indicate that our method can effectively circumvent occlusion for automated texture mapping, and generate 3D models with fewer obstructed textures, which largely improves the visual quality and realism of the models.

## 1. INTRODUCTION

A highly realistic city 3D model is the key foundation for the construction of a digital twin city (Li et al., 2022), especially the building 3D model plays an important role in many applications such as urban space simulation and urban refinement management (Haala and Kada, 2010; Wu et al., 2018). Compared with the traditional two-dimensional data, the realistic 3D model has incomparable advantages in expressing the actual geometry and spatial location of objects, etc. As the development of the city advances day by day, the overall spatial layout of the city is constantly changing from the original two-dimensional layout to the three-dimensional space, which also leads to the existence of different degrees of spatial occlusion between various objects in the city, in the complex urban scene, due to the air or ground image. In complex urban scenes, due to the natural limitation of non-contact remote acquisition methods or the objective influence of the complex environment around the building, the texture pictures of the 3D model of the building are often obscured by the surrounding trees or other objects, which cannot accurately portray the real surface texture information of the building and lead to texture mapping errors.

As the last part of 3D reconstruction, high-definition texture mapping, as an important means to improve the quality of building 3D models, directly affects the visualization effect and realism of building 3D models. Considering that point clouds are widely used in 3D reconstruction, they are also easy to obtain. Both the 3D laser scanning point clouds based on the laser scanner (Niemeyer et al., 2014) and the dense matching point clouds (Rhee and Kim, 2016) based on the image generation are more mature

in their acquisition and production methods. The point clouds record the 3D coordinate information of all points on the surface of the object in a unified coordinate system, which expresses the spatial geometry of the object more completely, and the texture mapping process can be optimized by the point clouds.

To address the occlusion problem of texture mapping of building 3D models, in this paper, a method to determine the occlusion relationship between building models and texture candidate images using 3D point clouds is proposed. First, the resolution-adaptive voxel occlusion generation method is proposed to voxelize the point cloud by calculating the point cloud voxel resolution with the number of point clouds and the enclosing box information; then, the occlusion judgment and image filtering method based on the spatial through-view relationship is proposed to filter the images based on the three-dimensional spatial through-view relationship among the image projection center, model and voxel; finally, the Markov random field is optimized based on the occlusion function. Finally, the data items of the Markov random field are optimized based on the occlusion function, and the most suitable image is selected for each facepiece of the model for texture mapping. This results in a more realistic 3D model.

The rest of this paper is organized as follows. Section 2 briefly reviews the existing texture mapping and its methods for circumventing occlusion. The proposed method and its key steps are described in detail in Section 3. Section 4 conducts experiments using different platforms or multi-platform fused images, laser point clouds, and image dense matching point clouds, and performs qualitative and quantitative cross-sectional comparisons with existing texture mapping algorithms. Conclusions and discussions are drawn in Section 5.

\* Corresponding author

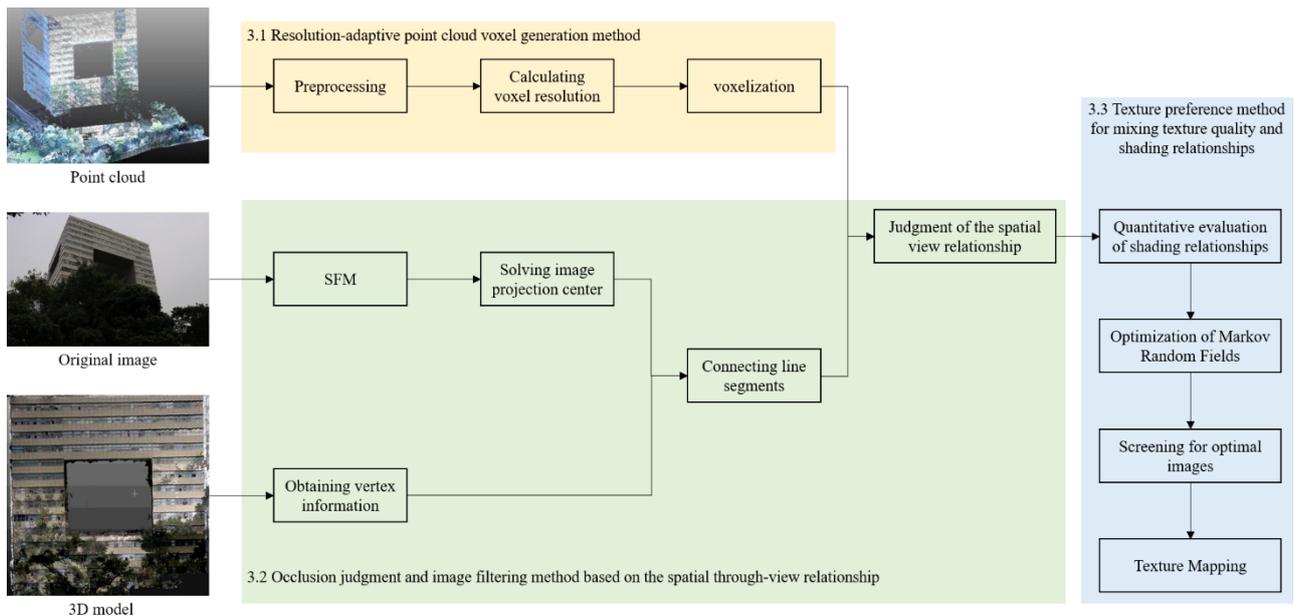


Figure 1. Workflow of our method.

## 2. RELATED WORK

With the progress of modern science and technology, photogrammetry and computer vision have been developed rapidly, and the application of real-world 3D models has become more and more widespread. Image-based 3D reconstruction can be understood as four algorithmic steps: Structure from Motion, Multi-View Stereo, surface structure network reconstruction (Snavely et al., 2006), and texture mapping. Numerous scholars have also designed different 3D reconstruction algorithms according to the applicability of each step algorithm, such as VisualSfM (Wu and Ieee, 2013), Bundler (Snavely, 2010), MVE (Fuhrmann et al., 2015), etc.

Texture mapping, as the most available step of 3D reconstruction, is one of the most widely used key techniques in computer vision and photogrammetry. The idea of texture mapping was first introduced by Catmull back in 1974 (Catmull, 1974). After that, Sinha designed an interactive system to texture large 3D models with flat surfaces (Sinha, 2008), and Tan et al. also proposed an interactive approach and focused on texture mapping of building surfaces (Tan et al., 2008). By applying pairwise Markov random fields to texture mapping, a view can be selected for each model slice, and its data items are used to determine the quality of the texture view (Lempitsky et al., 2007), so by optimizing the data items, there are different image filtering effects (Gal et al., 2010), either using normal vector angles or using projected gradient information (Waechter et al., 2014).

Early methods of texture mapping to circumvent occlusion relied on user interaction to mark the occluded objects (Sinha, 2008).

As the research on automated methods progressed, there were two main methods to circumvent occlusion, one based on image restoration and filtering, and one based on image and model 3D relationships for judgment. Some of the image-based methods are iterative screening by color information in the image (Grammatikopoulos et al., 2012), some are detecting duplicate regions for substitution (Zhou et al., 2016), and some are based on deep learning methods for masking detection (Ronneberger et al., 2015) and later repairing the masked regions (Yu et al., 2019; He et al., 2021). The substitution or repair methods do not essentially solve the masking problem, and the effect is different from the actual one. Judgment methods based on image and model 3D relationships often map images of geometrically missing objects

onto the surrounding model because the geometry of the model reconstruction is not complete enough.

To verify the effectiveness of the segmentation method proposed in this paper, an RF classifier is used to distinguish the classes of the clusters utilizing five kinds of features.

## 3. METHODOLOGY

The method in this paper is divided into three main steps. The first step is the Resolution-adaptive point clouds voxel generation method. The second step is the Occlusion judgment and image filtering method based on the spatial through-view relationship. The final step is the Texture preference method for mixing texture quality and shading relationships. The specific process is shown in Figure 1.

### 3.1 Resolution-adaptive point clouds voxel generation method

**3.1.1 Point clouds preprocessing.** Both laser point clouds and image-dense matching point clouds often cause data redundancy in the data acquisition process, so the point clouds data need to be pre-processed. For point clouds with large coverage, it is necessary to segment them to obtain the point clouds within the experimental range, and then it is necessary to align the point clouds data with the 3D model that needs texture mapping to the same coordinate system to judge the 3.2 spatial through-view relationships. For dense point clouds data, noise reduction and downsampling are required. Noise reduction can better represent the object geometry by eliminating discrete points, and downsampling can effectively improve the efficiency of the algorithm by reducing the number of point clouds without changing the geometry.

**3.1.2 Point clouds voxelization.** In this paper, we design a resolution-adaptive voxel occlusion generation method by solving the enclosing box of the pre-processed point clouds data, combining it with the point clouds density, calculating the adaptive voxel resolution, and spatially dividing the point clouds data based on it, traversing all voxels to see whether they contain point clouds data, extracting all valid voxels into a set, which can effectively express the morphology of the point clouds data. Calculate the Bounding Box corresponding to the original point clouds data based on the 3D spatial coordinate data of each point

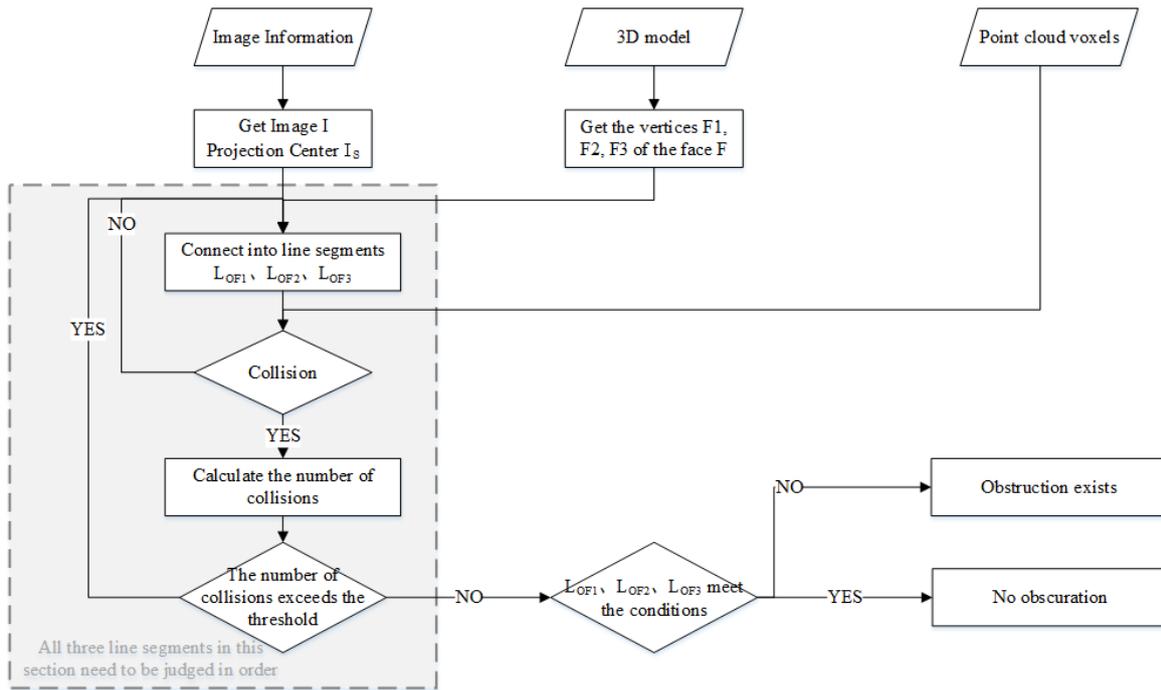


Figure 2. Technical route of spatial through-view relationship judgment

in the point clouds data, and obtain the maximum and minimum values of the point clouds in the three directions of X, Y, and Z axes as  $X_{max}$ ,  $X_{min}$ ,  $Y_{max}$ ,  $Y_{min}$ ,  $Z_{max}$ ,  $Z_{min}$ . The average Euclidean distance of the point clouds data in the three directions of X, Y and Z axes is calculated from the known extreme points.

$$\begin{cases} \Delta x = X_{max} - X_{min} \\ \Delta y = Y_{max} - Y_{min} \\ \Delta z = Z_{max} - Z_{min} \end{cases} \quad (1)$$

The volume  $V$  of the point clouds can be calculated from the average Euclidean distance in the three directions:

$$V = \Delta x * \Delta y * \Delta z \quad (2)$$

Through the obtained point clouds, the number of points  $N$  in it can be obtained, and the voxel resolution of this point clouds data can be solved by the number of points and the volume of the point clouds as  $PC_{resolution}$ .

$$PC_{resolution} = \frac{V}{N} = \frac{(\Delta x * \Delta y * \Delta z)}{N}, \quad (3)$$

$(N = 0, 1, 2 \dots \dots, n);$

This method can effectively calculate the adaptive voxel resolution, and the spatial partitioning of the point clouds data can be performed according to the adaptive resolution. The number of voxels in each axis can be calculated based on the adaptive voxel resolution and the extreme points in the three axes,  $X_{size}$ ,  $Y_{size}$ ,  $Z_{size}$  are the number of voxels in X, Y, and Z axes respectively.

$$\begin{cases} X_{size} = [\Delta x / PC_{resolution}] \\ Y_{size} = [\Delta y / PC_{resolution}] \\ Z_{size} = [\Delta z / PC_{resolution}] \end{cases} \quad (4)$$

Get the number of voxels in X, Y, and Z directions to solve for the number of voxels in the whole point clouds  $BBX_{size}$ .

$$BBX_{size} = X_{size} * Y_{size} * Z_{size} \quad (5)$$

The voxel resolution divides the enclosing box between the X, Y, and Z axes to get all voxels of the point clouds, and stores all voxels into the same set. By traversing all voxels to see if they contain point clouds data, all voxels containing point clouds are filtered out, and the voxel set is the result of point clouds voxel talk, which expresses the geometry of the original large number of point clouds by fewer voxels.

### 3.2 Occlusion judgment and image filtering method based on the spatial through-view relationship

The 3D reconstruction technique of image can obtain the information of internal and external orientation elements of each image in the process of SFM solution, mainly including the focal length, projection center coordinates, rotation matrix parameters and other related information of the image, which can be used to determine the position pose of the image in the spatial right angle coordinate system. By parsing the data generated in SFM, the projection center  $I_0(X_0, Y_0, Z_0)$  corresponding to each image  $I$  is extracted. The mesh model records the 3D spatial coordinate information of all points in the model, and the index information of every three neighboring points can constitute a triangular surface slice, based on which the position data of points and surfaces in the mesh model can be extracted by The three vertices corresponding to  $F$  are  $F_1(X_{F1}, Y_{F1}, Z_{F1})$ ,  $F_2(X_{F2}, Y_{F2}, Z_{F2})$ ,  $F_3(X_{F3}, Y_{F3}, Z_{F3})$ . Generating three line segments  $L_{OF1}$  in three-dimensional space by connecting three vertices  $F_1(X_{F1}, Y_{F1}, Z_{F1})$ ,  $F_2(X_{F2}, Y_{F2}, Z_{F2})$ ,  $F_3(X_{F3}, Y_{F3}, Z_{F3})$  of a single face sheet to the projection center  $I_0$  corresponding to each image  $I$ .

$$\frac{x-X_0}{x-X_{F1}} = \frac{y-Y_0}{y-Y_{F1}} = \frac{z-Z_0}{z-Z_{F1}} \quad (6)$$

$x \in (X_0, X_{F1}), y \in (Y_0, Y_{F1}), z \in (Z_0, Z_{F1})$

According to this method, the spatial position coordinate information between  $I_0(X_0, Y_0, Z_0)$  and  $F_2(X_{F2}, Y_{F2}, Z_{F2})$ ,  $F_3(X_{F3}, Y_{F3}, Z_{F3})$  is known, and also the equations of linear equations of line segments  $L_{OF2}$  and  $L_{OF3}$  can be solved.

The existence of an intersection between a line segment and a voxel is determined by performing a depth conflict check of the three-line segments connected to a cube formed by the eight vertices  $R_1(x_1, y_1, z_1)$ ,  $R_2(x_2, y_2, z_2)$ ,  $R_3(x_3, y_3, z_3)$ ,  $R_4(x_4, y_4, z_4)$ ,  $R_5(x_5, y_5, z_5)$ ,  $R_6(x_6, y_6, z_6)$ ,  $R_7(x_7, y_7, z_7)$ ,  $R_8(x_8, y_8, z_8)$  of each voxel in the six planes formed by that voxel.

$$\begin{cases} a_1x_1 + b_1y_1 + c_1z_1 + d_1 = 0 \\ a_1x_2 + b_1y_2 + c_1z_2 + d_1 = 0 \\ a_1x_3 + b_1y_3 + c_1z_3 + d_1 = 0 \end{cases} \quad (7)$$

Based on this equation the equation formula for the individual planes of voxels can be solved.

$$\begin{cases} a_1x + b_1y + c_1z + d_1 = 0 \\ \min \{x_1, x_2, x_3, x_4\} < x < \max \{x_1, x_2, x_3, x_4\} \\ \min \{y_1, y_2, y_3, y_4\} < y < \max \{y_1, y_2, y_3, y_4\} \\ \min \{z_1, z_2, z_3, z_4\} < z < \max \{z_1, z_2, z_3, z_4\} \end{cases} \quad (8)$$

With the known information on the eight vertices of the voxels, it is then possible to solve the equation formula for the six planes.

If there is an intersection between the line segment and the voxel surface piece, it is recorded as a collision and the position information of the intersection in the three-dimensional space is calculated as  $S_1(X_{S1}, Y_{S1}, Z_{S1})$ .

$$\begin{cases} a_1X_{S1} + b_1Y_{S1} + c_1Z_{S1} + d_1 = 0 \\ \frac{X_{S1}-X_0}{X_{S1}-X_{F1}} = \frac{Y_{S1}-Y_0}{Y_{S1}-Y_{F1}} = \frac{Z_{S1}-Z_0}{Z_{S1}-Z_{F1}} \end{cases} \quad (9)$$

When all voxels are traversed, the total number of collisions is recorded as  $S$  times, and the intersection points are  $S_n$   $n=0,1,2 \dots n$ . The overall process of spatial through-view relationship determination is shown in Figure 2.

After iterating through all the point clouds voxels, a threshold is set to determine the spatial relationship between the center of the image projection, the vertices of the model surface, and the point clouds voxels. According to the application in practice, the threshold value is set to 2 (because the point clouds and the model have been aligned to the same coordinate system, the voxel generated based on the point clouds contains all the point clouds inside the voxel, so the 3D model is also contained inside the voxel, when by connecting the model vertex and the image center into a line segment, the line segment will definitely pass through the voxel where the model vertex is located and produce the intersection, the collision count is 1, but when (When there is an occlusion between the model triangle vertices and the image center, the voxel generated by the point clouds of the occlusion will also collide with the line segment, which leads to a collision count greater than or equal to 2, so the threshold value is set to 2), when the collision count is less than the threshold value, it means that the image projection center and the model vertices maintain a through-view relationship, no occlusion; when the collision count is greater than the threshold value, it means that the image projection center and the model vertices cannot maintain a through-view relationship, then there is an occlusion between them, and there is an occlusion.

### 3.3 Texture preference method for mixing texture quality and shading relationships

**3.3.1 Occlusion function.** From the perspective of texture mapping in 3D reconstruction, occlusion is manifested as the existence of another model between the two when a model is observed from the direction of the field of view, resulting in a wrong texture mapping of the former model to the model observed in the back, thus producing a wrong texture mapping. A depth conflict check is performed between these three, image, model, and point clouds voxel, based on the spatial location, and a judgment of occlusion is made based on the results of spatial passages to determine whether there is occlusion between the image and this triangular surface sheet, but this is only a qualitative description, and a quantitative evaluation of each image is needed. Here, let the quality value of each image be  $V_{Ii}$  and the quality value of each model triangular surface sheet be  $V_{Fj}$ . Based on the calculation result of the spatial pass-view relationship, if the number of collisions  $S < 2$ , then let.

$$V_{Ii} = V_{Fj} \quad (10)$$

The equation expresses that when the number of collisions between the image projection center and the connection line

break of each vertex of the model triangular facet sheet and the point clouds voxel is less than 2, the mass value of the image is equal to the mass value of the triangular facet sheet, which is mathematically expressed as a one-to-one correspondence mapping relationship between the two. In this paper, we evaluate the occlusion status of each image relative to the model surface slice by setting a new function. Considering that the actual occlusion status exists only in two cases, with and without occlusion, and any degree and proportion of occlusion can be regarded as occlusion, we set the occlusion function as  $E_{Cover}$ , and its specific formula is expressed as follows.

$$E_{Cover} = [V_{Ii} = V_{Fj}] \quad (11)$$

The  $[\ ]$  represents the Iverson bracket, whose operation process is expressed as 1 if the condition inside the bracket is satisfied and 0 if it is not, which can effectively convert the Boolean value to an integer value. The expression can be interpreted as  $E_{Cover}$  is 0 if there is no occlusion between image  $I_i$  and surface  $F_j$ , and  $E_{Cover}$  is 1 if there is an occlusion between image  $I_i$  and surface  $F_j$ . The value of  $E_{Cover}$  quantitatively evaluates the occlusion relationship from a mathematical point of view.

**3.3.2 Markov Random Field Optimization.** The Markov random field represents the set of all objects that are only related to the properties of neighboring objects and not to the properties of objects in other regions. Based on the Markov random field concept, some scholars applied it to the work of texture mapping, where Lempitsky and Ivanov used the Markov random field to select a view for each triangular facet, the data term to determine the quality of the texture view, and the smoothing term to model the severity of the joints between the texture facets for smoothing, whose expressions are shown as follows.

$$E(M) = E_Q(M) + \lambda E_S(M) \quad (12)$$

M. Waechter, N. Moehrl et al. also proposed to fuse normal vector angle information and projected gradient information in the original basic data term proposed by Lempitsky for processing.

$$\begin{cases} E_Q(M) = E_{Data} \\ E_{Data} = - \int_{\phi(F_i, l_i)} \|\nabla(I_{li}(p))\|_2 dp \end{cases} \quad (13)$$

This method uses the Sobel operator to project the triangular surface slice  $F_i$  onto the image and calculates the gradient magnitude  $\|\nabla(I_{li}(p))\|_2$  by summing all pixels of the gradient magnitude image within the projection  $\phi(F_i, l_i)$  of  $F_i$ . In this paper, we combine the previous occlusion status of each image and the occlusion function  $E_{Cover}$  used for evaluation to judge the quality of the texture view of the data item is further optimized by adding the occlusion function  $E_{Cover}$  to the original one, and the optimized expression of the data item is:

$$E_Q(M) = E_{Data} + E_{Cover} \quad (14)$$

By further optimizing the data items of the Markov random field and adding the occlusion function  $E_{Cover}$ , the original data item  $E_{Data}$  is filtered by the information of normal vector angle and projection gradient, and  $E_{Cover}$  is filtered by the occlusion condition of the image, and the most suitable and optimal image is filtered for texture mapping of this triangle face slice. The other triangular slices are iterated using the same method so that each triangular slice has a texture image corresponding to it, and the neighboring slices using the same image can calculate the texture coordinate value of their vertices in the image so that their corresponding texture slices are also merged, which effectively reduces the time for texture seam processing afterward, and then the texture is leveled and color graded to complete the whole process of texture mapping. The whole process of texture mapping is completed, and a highly realistic 3D model is output.

Experiments	Data sources	Image Platform	Number of images	Model vertices	Point clouds type
No. 1	Huxin Island in Jiaxing City, China	Aerial platform	32	490173	laser point clouds and dense matching point clouds
No. 2	Shenzhen University, China,	Ground platform	14	127006	laser point clouds
No. 3	Dortmund(ISPRS)	Air-ground platform integration	41	358609	dense matching point clouds

**Table 1.** Statistics of data used in the experiment

#### 4. EXPERIMENTS AND ANALYSIS

##### 4.1 Test datasets

To verify the effectiveness of the proposed method, this paper uses the public test datasets published by the International Society for Photogrammetry and Remote Sensing (ISPRS) and several urban area datasets collected by ourselves to conduct comparative experiments, mainly involving different platforms or multi-platform fused images, laser point clouds and image dense matching point clouds for experiments, and qualitative and quantitative cross-sectional comparisons with existing texture mapping algorithms.

In the first set of experiments, 32 aerial platform images of the Huxin Island scenic area in Jiaxing City, Zhejiang Province, China, were acquired using DJI Phantom IV RTK, and dense matching point clouds based on images and laser point clouds acquired by GeoSlam scanning were used for the experiments, respectively, and the to-be-textured mapping models for the experiments were generated by existing 3D reconstruction algorithms.

The second set of experiments used 14 images of the ground platform of the Science and Technology Building of Shenzhen University, China, captured by a Canon EOS80D DSLR camera. The point clouds in the experiments are laser point clouds acquired by GeoSlam scanning, and the 3D models are generated by existing 3D reconstruction algorithms.

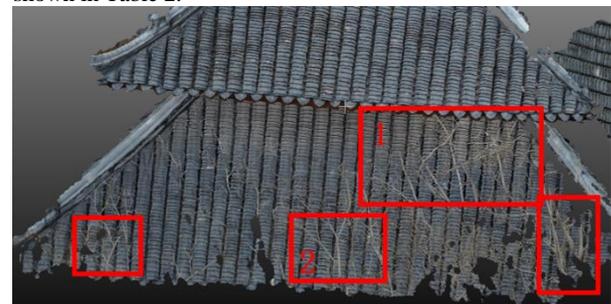
The third set of experiments used the Dortmund, Germany images released by the International Society for Photogrammetry and Remote Sensing (ISPRS), from which 20 images of the ground platform acquired by Sony Nex-7 and 21 images of the air platform acquired by the multi-rotor aircraft DJI S800 with Sony Nex-7 were selected, a total of 41 images, and the point clouds were dense matching point clouds generated from these 41 images, and the 3D model generated by the existing 3D reconstruction algorithm. The specific experimental data are shown in Table 1.

##### 4.2 Results for the Jiaxing dataset

In Experiment 1, the experiment is first conducted by using six images captured by DJI Genie 4 RTK UAV. Based on the six images, their camera pose and orientation element information inside and outside the image are recovered by SFM, and then the dense point clouds of the scene are output by dense reconstruction of MVS, which has 95850883 dense point clouds. The occluded voxels are generated using the pre-processed dense matching point clouds, and the most suitable image is selected to complete texture mapping based on the spatial flux and mapping relationships among the image, model, and voxels. The comparison of the results of the method in this paper with the open-source algorithm mvs-texturing is shown in Fig. 3.

The red box area indicates the part of the open-source algorithm mvs-texturing texture mapping where there is still occlusion, and the textures such as tree branches are mapped to the tile model of the gazebo, and the green box part indicates the result after texture mapping at the same location using the method of this paper. By comparing the visualization of the 3D models after te

xture mapping of the two methods, the method of this paper is better than the mvs-texturing method. The two-dimensional images were obtained by a camera at the same camera angle as the models generated by the two methods, and the image pixels were 466\*219, two ranges were selected for region one and region two in Figure 3, and there was a certain degree of occluded texture in the regional range, and the pixels of region one were 130\*70 and the pixels of region two were 75\*43. The details are shown in Table 2.



(a) Results of mvs-texturing



(b) Results of our method

**Fig. 3** Comparison of the results of the mvs-texturing and our method in Experiment 1

Region	occluded condition	mvs-texturing	our method
Region 1	Number of occluded pixels	3864	461
	Total percentage of occluded pixels	42.46%	5.07%
Region 2	Number of occluded pixels	893	56
	Total percentage of occluded pixels	27.69%	1.74%

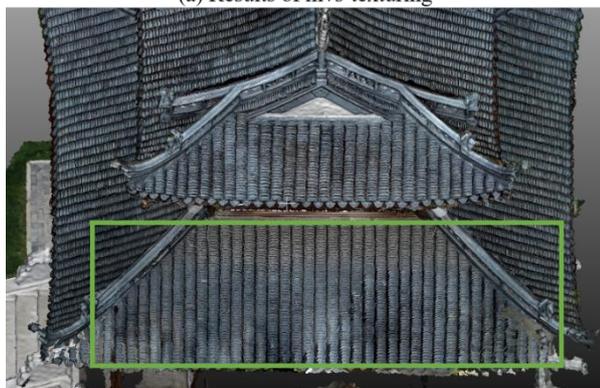
**Table 2** Occlusion status statistics after texture mapping using dense point clouds in Experiment 1

According to the counted number of occluded pixels, the number of mvs-texturing method is 3864 and 893 in the two regions after mapping, while the number is 461 and 56 after mapping by the method of this paper, the number of occluded pixels is greatly reduced, and the percentage of occlusion is also reduced from 42.46% and 27.69% to 5.07% and 1.74%, so the

effect of circumventing occlusion is greatly improved. The effect of occlusion avoidance has been greatly improved. The above experiments are conducted using image-based dense matching point clouds, followed by experiments using laser point clouds captured by a GeoSlam handheld laser scanner. The experiments are conducted using 32 images captured by DJI Phantom 4 RTK and their corresponding image information, laser point clouds data, and the model to be texture mapped.



(a) Results of mvs-texturing



(b) Results of our method

Figure 4 Comparison of texture mapping results using laser point clouds in Experiment 1

The mvs-texturing texture mapping method still cannot circumvent the occlusion, and the texture image of the tree branch is still incorrectly mapped on the model texture, while in the experiments based on the method of this paper, the spatial through-view relationship judgment and depth conflict checking can effectively circumvent the occlusion by voxelizing the laser point clouds data, and the two-dimensional images are obtained by a camera at the same camera angle for the models generated by the two methods, and the acquired image pixels are  $486 \times 314$ , and all the occluded pixels in the images are counted, and the details are shown in Table 3.

	mvs-texturing	our method
Number of occluded pixels	12764	169
Total percentage of occluded pixels	8.36%	0.11%

Table 3 Statistics of occlusion condition after texture mapping using laser point clouds in Experiment 1

According to the number of occluded pixels in the whole image, the number of the mvs-texturing method is 12,764 after mapping, and the number is 169 after mapping by the method of this paper, and the number of occluded pixels is greatly reduced; in the comparison of the percentage of occluded pixels, the percentage of mvs-texturing method is 8.36%, and the percentage of this paper is 0.11%.

pixels, the percentage of mvs-texturing method is 8.36%, and the percentage of this method is 0.11%.

#### 4.3 Results for the Shenzhen University dataset

In Experiment 2, 14 images of the Science and Technology Building of Shenzhen University on the ground platform were acquired using a Canon EOS80D DSLR camera, while laser point clouds data within the survey area were acquired using a GeoSlam handheld laser scanner. By inputting the images and their related information, the laser point clouds data and the model are to be texture mapped for the related experiments.



(a) Results of mvs-texturing (b) Results of our method  
Figure 5 Comparison of the results of the mvs-texturing and our method in Experiment 2

The mvs-texturing method in which a large number of tree foliage textures are incorrectly mapped to the side of the building at the rear, while using the method in this paper effectively circumvents the tree foliage occlusion in texture mapping, by acquiring a two-dimensional image by a camera at the same camera angle for the models generated by both methods, the acquired image pixels are  $445 \times 446$ , and all the occluded pixels in the image are counted, as shown in Table 4.

	mvs-texturing	our method
Number of occluded pixels	82469	112
Total percentage of occluded pixels	41.55%	0.06%

Table 4 Statistics of occlusion condition after texture mapping using laser point clouds in Experiment 2

According to the number of occluded pixels in the whole image, the number of the mvs-texturing method is 82,469 after mapping, and the number is 112 after mapping by the method of this paper, and the number of occluded pixels is greatly reduced; in the comparison of the percentage of occluded pixels, the percentage of mvs-texturing method is 41.55%, and the percentage of this paper is 0.06%. In the comparison of the percentage of occluded pixels, the percentage of mvs-texturing method is 41.55%, and the percentage of this paper is 0.06%, the canopy texture on the building façade is well eliminated in the screening process.

#### 4.4 Results for the Dortmund dataset

In Experiment 3, 20 images of the ground platform and 21 images of the air platform from the public dataset-Dortmund Experiment dataset provided by ISPRS were used, as well as the dense matching point clouds generated based on these 41 images, and the results of the experiment are shown in Figure 6.

The mvs-texturing method retains a large number of erroneous tree textures on the mapped building surface, while the method in this paper effectively removes them by acquiring two-dimensional images from the models generated by both methods at the same camera angle, and the acquired image pixels are  $490 \times 554$ , and all the occluded pixels in the images are counted, as shown in Table 5.

According to the number of occluded pixels in the whole image, the number of mvs-texturing method is 81464 after mapping,

and the number of mvs-texturing method is 511 after mapping by this paper, the number of occluded pixels is greatly reduced, but some wrong textures still remain in the upper right of the building. In the comparison of the percentage of occluded pixels, the percentage of the mvs-texturing method is 30.01%, and the percentage of this paper is 0.19%, so the effect of circumventing occlusion has been greatly improved.



(a) Results of mvs-texturing (b) Results of our method  
Figure 6 Comparison of the results of the mvs-texturing and our method in Experiment 3

	mvs-texturing	our method
Number of occluded pixels	81464	511
Total percentage of occluded pixels	30.01%	0.19%

Table 5 Occlusion status statistics after texture mapping using dense matching point clouds in Experiment 3

#### 4.5 Experimental analysis

##### 4.5.1 Validity of the method

Based on the visualization of the experiments in this paper shown in Figs. 3 to 6, the method in this paper achieves an automated texture mapping to circumvent occlusion, which is a great improvement in the effect, whether it is the images collected independently or the public dataset provided by ISPRS - Dortmund experimental dataset, or the smallest treetop branches or the trunk canopy of the whole tree, which can be effectively removed, thus making the model more realistic and better visualized.

##### 4.5.2 Effect of image platform sources

According to the analysis of the image data used in the three sets of experiments, experiment 1 used 32 images collected by the UAV on the air platform, experiment 2 used 14 images collected by the Canon EOS80D DSLR camera on the ground platform, and experiment 3 selected 20 images from the ground platform and 21 images from the air-ground platform in the public dataset - Dortmund Experimental dataset provided by ISPRS. In Experiment 3, 20 ground platform images and 21 air platform images were selected from the public dataset provided by ISPRS - Dortmund Experimental Dataset for the fusion of air-ground platforms. The results of multiple comparison experiments show that the images acquired by the air platform alone, the images acquired by the ground platform, or the images fused by the air-ground platform can be used to complete the automatic occlusion avoidance texture mapping of buildings based on the method in this paper, and the method designed in this paper has no limitation on the source of the images, which can achieve good effect of occlusion avoidance.

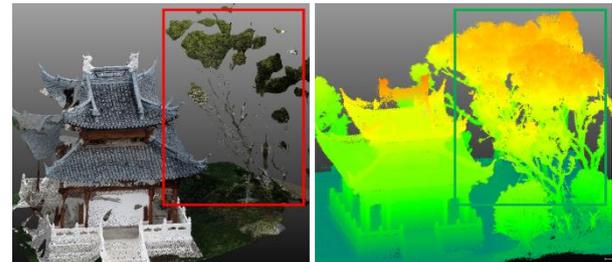
##### 4.5.3 Effect of different point clouds sources

According to the analysis of the experiments based on the point clouds data used in the three groups of experiments, there are two types of point clouds used in the experiments of this paper: laser point clouds and dense matching point clouds. There are

two groups of experiments in which laser point clouds are used. In experiment 2, the percentage of occluded texture decreased from 41.55% to 0.06% after using the laser point clouds; in experiment 1, the percentage of occluded texture decreased from 8.36% to 0.11% after using the laser point clouds, and the effect of avoiding occlusion was significant, and the percentage of occluded texture decreased significantly.

There are two sets of experiments using dense matching point clouds. Experiment 3 decreased the occluded texture ratio from the original 30.01% to 0.19% after the experiment using the public dataset provided by ISPRS - Dortmund experimental dataset, and Experiment 1 conducted the related test experiment after generating the dense matching point clouds using the autonomously acquired image data, the two statistical regions in the experiment were reduced from the original 42.46% and 27.69% to 5.07% and 1.74%, and although the percentage of occlusion decreases, the decrease is smaller compared to the two experiments conducted with laser point clouds and the experiments using dense matching point clouds in Dortmund. The visualization effect contains the presence of some of the more obvious occluded textures.

By comparing the experimental data, since only 6 images were used for dense reconstruction in Experiment 1, the generated dense point clouds did not represent the object geometry well, and the dense matching point clouds used in Experiment 1 were compared with the laser point clouds morphology as shown in Figure 7. When more images are involved in the dense reconstruction, such as in Experiment 3, where 41 images of the air-ground platform are fused to generate the dense matching point clouds, the experimental results are significant in avoiding occlusion.



a Dense matching point clouds b Laser Point clouds  
Figure 7 Comparison of dense point clouds and laser point clouds morphology in experiment 1

## 5. CONCLUSION

In this paper, in order to address the occlusion problem existing in the texture mapping of building 3D models, a method for judging the occlusion relationship between building models and texture candidate images using 3D point clouds is proposed. First, the resolution-adaptive voxel occlusion generation method is proposed, which voxelized the voxel-based on the adaptive voxel resolution of the point clouds calculation; then, the occlusion judgment and image filtering method based on the spatial through-view relationship is proposed, which screens the images based on the spatial through-view relationship among the image projection center, model vertices, and voxels; finally, the data items of the Markov random field based on the occlusion function are. Finally, the data items of the Markov random field are optimized based on the occlusion function, and the most suitable image is selected for each facepiece of the model for texture mapping. The main innovation of this method is to overcome the natural deficiency of current texture detection and repair of occluders directly from 2D images and to shift to a 3D spatial perspective. Experiments are conducted using three sets of data for quantitative and qualitative analysis to evaluate the proposed method, and the results show that different types of point clouds

ds data or images from different platform sources can achieve a good effect of occlusion avoidance, which can effectively support cross-platform multi-source image data and point clouds data. The related technologies can be widely used in smart cities, digital twins, and other fields to help the construction of realistic 3D models.

#### ACKNOWLEDGEMENTS

This study was supported by the National Natural Science Foundation of China (42001407, 41971341, 41971354), and the Guangdong Basic and Applied Basic Research Foundation (2019A1515110729, 2019A1515010748, 2019A1515011872). The authors would like to acknowledge the provision of the datasets by ISPRS and EuroSDR, released in conjunction with the ISPRS scientific initiative 2014 and 2015, lead by ISPRS ICWG I/Vb.

#### REFERENCES:

- Catmull, E., 1974. A Subdivision Algorithm for Computer Display of Curved Surfaces. The University of Utah.
- Fuhrmann, S., Langguth, F., Moehrle, N., Waechter, M. and Goesele, M., 2015. MVE—An image-based reconstruction environment. *Computers & Graphics*, 53: 44-53.
- Gal, R., Wexler, Y., Ofek, E., Hoppe, H. and Or, D.C., 2010. Seamless Montage for Texturing Models, *Computer Graphics Forum*.
- Grammatikopoulos, L., Kalisperakis, I., Karras, G. and Petsa, E., 2012. Automatic multi-view texture mapping of 3D surface projections.
- Haala, N. and Kada, M., 2010. An update on automatic 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6): 570-580.
- He, K. et al., 2021. Masked Autoencoders Are Scalable Vision Learners.
- Lempitsky, V., Ivanov, D. and Ieee, 2007. Seamless mosaicing of image-based texture maps, *IEEE Conference on Computer Vision and Pattern Recognition*. *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, pp. 829-+.
- Li, X. et al., 2022. Big data analysis of the Internet of Things in the digital twins of smart cities based on deep learning. *Future Generation Computer Systems*, 128: 167-177.
- Niemeyer, J., Rottensteiner, F. and Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87: 152-165.
- Rhee, S. and Kim, T., 2016. DENSE 3D POINT CLOUD GENERATION FROM UAV IMAGES FROM IMAGE MATCHING AND GLOBAL OPTIMIZATION. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B1: 1005-1009.
- Ronneberger, O., Fischer, P. and Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. Springer International Publishing.
- Sinha, S., 2008. Interactive 3D Architectural Modeling from Unordered Photo Collections. *ACM*.
- Snavely, N., 2010. Bundler: Structure from Motion (SfM) for Unordered Image Collections.
- Snavely, N., Seitz, S.M. and Szeliski, R., 2006. Photo tourism: Exploring photo collections in 3D., 25(3): 835-846.
- Tan, Y., Kwoh, L.K. and Ong, S.H., 2008. LARGE SCALE TEXTURE MAPPING OF BUILDING FACADES.
- Waechter, M., Moehrle, N. and Goesele, M., 2014. Let There Be Color! Large-Scale Texturing of 3D Reconstructions. Springer International Publishing, Cham, pp. 836-850.
- Wu, B., Xie, L., Hu, H., Zhu, Q. and Yau, E., 2018. Integration of aerial oblique imagery and terrestrial imagery for optimized 3D modeling in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139: 119-132.
- Wu, C.C. and Ieee, 2013. Towards Linear-time Incremental Structure from Motion. In: *W. Univ (W. Univ), International Conference on 3D Vision (3DV)*, Univ Washington, Seattle, WA, pp. 127-134.
- Yu, J., Lin, Z., Yang, J., Shen, X. and Huang, T., 2019. Free-Form Image Inpainting With Gated Convolution, 2019 *IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Zhou, G., Wang, Y., Yue, T., Ye, S. and Wang, W., 2016. Building Occlusion Detection From Ghost Images. *IEEE Transactions on Geoscience & Remote Sensing*, 55(2): 1074-1084.