Semantic segmentation guided feature point classification and seam fusion for image stitching

Huafeng Huang¹, Fei Chen¹, Hang Cheng¹, Liyao Li² and Meiqing Wang¹

Abstract
Image stitching can be employed to stitch images taken from different times, perspectives, or devices into a panorama with a wider view. However, the imaging specification of images to be stitched is strict. If the imaging specification is not satisfied, artefacts caused by inaccurate alignment and unnatural distortion will occur. Semantic segmentation can solve the classification problem at the pixel level; however, image stitching significantly depends on the accuracy of feature points. Therefore, this paper proposes an image stitching algorithm based on semantic segmentation to guide feature point classification and seam fusion. First, the images are recognized by a cascade semantic segmentation network, and the image feature points are classified. Thereafter, the corresponding homography transformations are calculated using different class feature points, and the best homography mapping for the entire target image is selected. Finally, a seam-cutting algorithm based on semantic segmentation is used to compute the seam, and a feathering Poisson fusion with distance transformation is used to eliminate artefacts and light differences. Experiments show that the algorithm can generate transitional natural and perceptual stitching results even under the influence of perspective and light differences.

Keywords
Semantic segmentation, image stitching, feature points, homography transformation, seam-cutting, Poisson fusion

Received 4 September 2021; Revised received 15 November 2021; accepted 22 November 2021

Introduction
The clarity of an image captured by a camera depends on the resolution of the camera. When capturing a scene with a large field of view, local clarity can decline. This limitation can be addressed through the image stitching algorithm,¹ ensuring that the stitched image can be obtained without reducing the resolution. Therefore, image stitching is widely used in aviation technology, virtual reality, machine vision, three-dimensional reconstruction, medical diagnosis, and other fields. Image stitching algorithms require the following processes: input reference image and target image, feature point detection, feature point matching, and screening,² calculation of the corresponding transformation, and then putting the transformed reference image and target image into the same canvas. Through the aforementioned processes, the image stitching algorithm can effectively stitch images that may originate from different devices, different perspectives, and different times into an image with a wider field of vision. Furthermore, it is difficult for users and other viewers to observe stitching traces.

However, the quality of the stitching algorithm determines the clarity and naturalness of the stitching results. If the corresponding transformation is not sufficiently accurate, it will blur or artefact the overlapping region. The simplest method for calculating the transformation of image stitching is to calculate the homography transformation corresponding to two images.³ However, in the case of a large parallax, the transformation error is large. If the

¹College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China
²Engineering Research Center for ICH Digitalization and Multi-source Information Fusion, Fujian Polytechnic Normal University, Fuqing, China

Corresponding author:
Hang Cheng, College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China.
hcheng@fzu.edu.cn
global feature points are used to calculate the global homography transformation, fuzzy artefacts in overlapping areas and serious perspective distortion in non-overlapping areas will simultaneously. Therefore, Gao et al.\textsuperscript{4} set a certain threshold by screening the feature points, dividing the feature points into the background planes points and the ground planes points, and then independently calculating the homography transformation of the two groups of feature points. Thereafter, they weighted the two homography transformations, which improved the alignment performance of overlapping regions and alleviated the perspective distortion of non-overlapping regions. Zaragoza et al.\textsuperscript{5} meshed the reference and target images and improved the alignment effect by independently calculating the homography transformation corresponding to each mesh. However, a mesh without feature points is distorted or deformed. When the parallax is large, the nearest feature point is used to calculate the homography transformation. However, the close feature points may belong to different objects, and the distant feature points are used in areas with few feature points.

To solve the aforementioned problems, a semantic segmentation guided image stitching algorithm was proposed in this paper.

The main contributions of this paper include three aspects:

1. Classify the feature points in combination with semantic segmentation to provide more accurate prior information for the alignment algorithms (as shown in Figure 1b and c).
2. Calculate the corresponding homography transformations for the class of feature points in each region to avoid being affected by the neighbouring different class feature points, and select the ideal homography transformations as the global homography transformations (as shown in Figure 1d).
3. A segmentation-aware seam algorithm is used to compute the seams of stitched images and an optimized Poisson fusion algorithm is used to fuse the images, which further improves the robustness of the algorithm to images with obvious light transformation (as shown in Figure 1e).

Related work

Image stitching\textsuperscript{6–13} The early image stitching algorithm used a global homography transformation to align two images; however, the algorithms had strict imaging requirements regarding the two images. Both images must rotate around the same projection centre and capture a plane. It is impossible to obtain a set of images that meet this criterion in real life. To compensate for this limitation, Gao et al. grouped the feature points, divided the image into two planes, and calculated the homography transformations of the two planes separately,\textsuperscript{4} which improved the alignment ability of the image stitching algorithm. This method improves the accuracy of homography transformation by improving the correspondence of feature points. Zaragoza et al.\textsuperscript{5} divided the images into equal grids and then calculated the corresponding homography transformations for each grid. The traditional algorithms consider the location of the feature points to some extent in calculating the homography transformation. Therefore, further classification of these feature points can further improve the accuracy of the alignment algorithm. To solve the aforementioned limitations, Lin et al.\textsuperscript{14} linearized the homography transformation of the grid on the basis of the dense mesh transformation of as-projective-as-possible (APAP),\textsuperscript{5} making the transition between grids smoother. The similarity transformation is introduced in the overlapping and the non-overlapping region of the target image to make the result more natural. Chang et al.\textsuperscript{15} divided the overlapping area into three areas. In the first area, only the homography transformation is used to align the target image and the reference image, and in the third area, only the global similarity transformation is used. In the middle of the two regions, the homography transformation will naturally transition to the similarity transformation. Li et al.\textsuperscript{16} use an elastic transformation to stitch images, which greatly improves the degree of freedom of alignment.

Seam-cutting and image fusion\textsuperscript{17–23} Stitching algorithms often cause artefacts in overlapping areas owing to parallax. Therefore, Gao et al.\textsuperscript{24} resolved this problem by dividing the overlapping area into two areas using a seam with minimal energy, where the pixels on one side of the seam originate from the reference image, and the pixels on the other side originate entirely from the target image. Li et al.\textsuperscript{25} designed a seam based on human perception, which avoids highly salient areas and improves both sides of the seam. The images to be stitched are taken at different times, perspectives, or even devices. Therefore, even if the alignment performance of the algorithm is strong, the stitching results may not be natural, and there will be differences in light or colour. Different fusion algorithms are used to solve this problem. Common fusion algorithms are alpha and linear fusion. Pyramid fusion uses different fusion rules to fuse different layers by extracting the features of the Gaussian or Laplace pyramids. The Poisson image fusion\textsuperscript{26} proposed by Patrick is based on the Poisson equation, making the edges of the fusion more natural and smooth.

Semantic segmentation and image stitching\textsuperscript{27–30} Semantic segmentation classifies every pixel in an image and is widely used in medical images, unmanned driving, and other fields. Dai et al.\textsuperscript{27} proposed cascading objects to achieve segmentation. In addition, Dai et al.\textsuperscript{28} explored cascading instance and object segmentation, and Zhao\textsuperscript{29}
introduced a multi-scale pyramid model to optimize semantic segmentation. Image stitching has strict requirements for the correspondence of feature points. Classifying feature points through semantic segmentation can improve the correspondence between the feature points. Thus, the homography calculation can be performed more accurately.

The algorithm proposed in this paper classifies feature points using semantic segmentation. The homography transformation of the feature points is calculated based on the class, and the best global homography transformation is then selected. Based on the semantic segmentation result, the optimal seam is computed for the stitching result; the image alignment performance and natural effect are further improved using the feathering Poisson fusion algorithm with distance transformation.

**Semantic segmentation guided feature point classification**

Semantic segmentation can provide pixel-level classification results for images. Inspired by the task of image semantic segmentation, the extracted feature points were classified by semantic segmentation, and the alignment model was solved based on the class of feature points, which improved the alignment ability. This section describes the cascading semantic segmentation module used in this study and classifies the feature points further using this module.

**Cascade semantic segmentation**

The deep learning method originally applied to the segmentation task is called image block classification, which uses
the image blocks around the pixel to classify the pixels. Fully convolutional networks (FCN)\textsuperscript{31} proposed an end-to-end convolution neural network structure to enable semantic segmentation to be conducted intensively without a full connection layer. Currently, the most popular structure of the semantic segmentation network is the encoder/decoder structure, which achieves an efficient classification by downsampling the input image to the low-resolution features. In addition, it restores these features to the full-resolution segmentation result using up-sampling. In this paper, we introduced a cascade segmentation network that combines background classes and object classes in a cascade manner to obtain finer segmentation results.

In a real scene, some background objects occupy a large part of the pixels, such as outdoor sky and buildings, indoor ceilings, floors, and walls. Therefore, the pixel proportion occupied by other small target objects is small, which affects the final segmentation result. Small objects are often objects of interest. However, most semantic segmentation algorithms ignore the spatial relationship between background classes and object classes, such as walls and windows. To alleviate the pixel distribution difference between the two classes, a cascade semantic segmentation network was proposed, which divides the network into background and object streams. Furthermore, training the background stream to obtain all background classes generates background segmentation and dense object mapping, which indicates the probability that a pixel is a member of an object in the foreground; object streams classify discrete objects, and further split each discrete object from the background stream. The result is combined with the result of the background segmentation to generate a scene semantically (as shown in Figure 2).

The proposed cascade segmentation module was integrated into Dilated-Net\textsuperscript{30,32} with the encoder ‘resnet101dilated’ and decoder ‘ppm-Deepsup’. The input image was sampled under three pooling layers with a downsampling rate of 8. After the third pooling layer, the background and object streams were trained separately using Dilated-Net, maintaining the spatial dimensions of the feature mapping of the two streams the same.

The background stream and object stream in the network can be trained end-to-end by sharing weights from the lower convolution layers. Each loss function was set for the two streams, and cross-entropy loss per pixel was used as the loss function for the background stream $L_{bg}$, which accumulates cross-entropy per pixel for all background classes:

$$L_{bg} = - \sum_{i=1}^{M} a_i \log (b_i)$$  

where $a_i$ is the ground truth class, $b_i$ is the class of prediction, and $M$ is the total number of background classes. Similarly, the cross-entropy loss per pixel was used for object classes $L_{ob}$. It accumulated cross-entropy per pixel for all output classes (all discrete object classes). Object mapping was provided as a binary mask to determine whether a pixel was a member of the background class or the object class. Therefore, $L_{ob} = - \sum_{i=1}^{N} a_i \log (b_i)$, where $N$ is the total number of object classes. The total training loss is

$$\text{Loss} = L_{bg} + L_{ob}$$  

The semantic segmentation network of the cascaded background stream and object stream was used to obtain the segmented results of the reference and target images, and it guided the feature points into different classes. In the following sections, we explained the classification of feature points using semantic segmentation results.

**Semantic classification of feature points**

Similar to most traditional stitching algorithms, the scale-invariant feature transform (SIFT)\textsuperscript{33} algorithm was used
to detect the feature points of the reference and target images, and then the matching algorithm was used to match the corresponding points of the two images.

By further classifying the feature points using the semantics segmentation results, the classes of the feature points were obtained, thus improving the alignment performance of the object class of interest as well as the alignment solution accuracy. However, the matching errors of feature points were further eliminated. If the matching points in the reference image and the target image belong to different classes, it indicates a matching error.

The location of the feature points detected by the scale-invariant feature transform (SIFT) \(^3\) algorithm was mapped to the semantics segmentation result image, and the corresponding classes of the feature points were obtained according to the red-green-blue (RGB) value of the semantics segmentation result (as shown in Figure 3).

**Image stitching model**

In the previous section, different classes of feature point groups were derived from the semantic segmentation networks. This section introduces how to use these different classes of feature point groups to obtain corresponding homography transformations and select the ideal alignment performance as the global homography transformations.

Because the target and reference images may originate from a different time, different perspectives, and different devices, the corresponding feature points were detected by the scale-invariant feature transform (SIFT) \(^3\) algorithm, filtered, and then classified by the semantic segmentation network in the previous section to obtain different classes of feature points from the target and reference images.

Two parallel lines can intersect in the image space at infinite distances, such as the ‘rail’ intersecting at a distance. Let \(\hat{q} = [x, y, 1]^T\) and \(\hat{p} = [x', y', 1]^T\) be the feature points in the target image and the reference image in homogeneous coordinates to satisfy this condition. Therefore, for a pair of matched feature points, the homography transformation is calculated as

\[
\hat{p} = H\hat{q}
\]

(3)

where \(H \in \mathbb{R}^{3 \times 3}\) is the homography transformation matrix. Therefore, at least four pairs of feature points were required.
to solve the corresponding homography transformation. For more feature point pairs of feature points, the solution formula provided in Zaragoza et al.\textsuperscript{5} was used to calculate the homography transformation corresponding to \( m \) classes of feature points obtained from the semantic segmentation network, \( m \) is the total number of feature point pairs in this category. We can obtain \( m \) homography transformations as follows:

\[
\mathbf{h}_j = \arg \min_{\mathbf{h}} \sum_{i=1}^{n} \| \mathbf{A}_i \mathbf{h} \|_2 = \arg \min_{\mathbf{h}} \| \mathbf{A} \mathbf{h} \|_2^2 \tag{4}
\]

where \( j = 1, 2, \ldots, m \) is the sequence number of the feature point class, \( n \) is the sequence number of the pair of feature point pairs, and \( \mathbf{A} \mathbf{A}^T = \sum_{i=1}^{n} \mathbf{A}_i \mathbf{A}_i \).

To better align the object classes in semantic segmentation, the homography transformation corresponding to the background classes was removed from the above \( m \) homography transformations to reduce the influence of the background homography transformation on the object class and improve the accuracy of object class alignment. Therefore, the transformation error of the homography transformation of each object class to all feature points is calculated as follows:

\[
\epsilon_{\mathbf{h}} = \sum_{i=1}^{k} \| \mathbf{h}_i \mathbf{p}_i - \mathbf{q}_i \|_2^2 \tag{5}
\]

where \( k \) represents the number of all feature-point pairs. The object class homography transformation with the smallest error was selected as the ideal global homography transformation. Through the above steps, the accurate alignment of all feature points was further realized on the premise of effective alignment of object classes in the segmentation results.

**Semantic segmentation perception-based seam fusion**

In the previous section, the results of semantic segmentation and homography transformation can make the regions with feature points coincide well; however, perfect alignment cannot be achieved in the local regions without feature points, and even artefacts appear. Therefore, this section introduces a method to facilitate semantic segmentation results and further improve the stitching results.

**Seam construction via semantic segmentation perception**

To further reduce the artefacts in overlapping areas, we set up a semantic segmentation perception seam, which calculates the ideal seam according to the performance based on the semantic segmentation results. The principle of the algorithm is shown in Figure 4. The blue and yellow areas are the non-overlapping areas of the reference and target images, and the green area represents the overlapping area of the two images. The task of seam-cutting was to ensure that the label \( l_p \) on the one side of the overlapping area originates from the reference image, and the other side originates from the target image.

The semantic segmentation perception-based seam algorithm defines a seam energy function composed of a pixel set \( P \), which is expressed as

\[
E = \sum_{p \in P} E_d(p, l_p) + \sum_{(p,q) \in N} \delta(p, q)E_s(p, q, l_p, l_q) \tag{6}
\]

where \( N \) represents a four-connected neighbourhood. The first term of the energy function \( E_d(p, l_p) \) is the data cost of assigning label \( l_p \) to pixel \( p \), where \( E_d(p, l_p) = -\nabla I_{l_p} \), and the second term represents the smoothing cost of label \( l_p \) and label \( l_q \), respectively, assigned to pixel \( p \) and pixel \( q \) in the four-connected neighbourhood, where \( E_s(p, q, l_p, l_q) = |l_p - l_q| (D_s(p) + D_s(q)) \). Therefore, when the two-pixel labels in the connected neighbourhood are the same, the smoothing cost is zero; otherwise, the smoothing cost is \( D_s(p) + D_s(q) \), where \( D_s = \varepsilon(\|l_0 - l_i\|_2^2 + 2\|l_0 - \nabla I_{l_i}\|_2^2) \) is the sum of the gradient difference of the pixel position in the reference image and the target image, \( \varepsilon(x) = 1 + \{1/[1 + e^{-4\kappa(x-\gamma)}]\} \) is used to determine whether the colour difference is visible, and \( \gamma \) is the colour difference. When it is less than \( \tau \), the colour difference is invisible; when it is greater than \( \tau \), it is visible. Parameter \( \kappa = 1/0.06 \), which is determined by the maximum inter-class variance.

Using the above formula, the seam can successfully bypass the misaligned area. However, research shows that the sensitivity of the human eye to different objects is different; the human eye always focuses on the highly salient area. Therefore, a seam that conforms to the human eye should avoid the salient area.

To achieve the above purpose, combined with the results of semantic segmentation, the average pixel significance \( \delta(\cdot) \) was used to define the salient weight:

\[
\delta(p, q) = \begin{cases} 
0, & \text{if } p \text{ or } q \in \text{background} \\
\frac{\delta(p) + \delta(q)}{2}, & \text{otherwise}
\end{cases} \tag{7}
\]

According to the semantic segmentation result, the smoothing cost is zero if the seam passes through the background class defined in the segmentation result. Through this salient weight constraint, the smoothing cost is weighted to obtain a new seam energy function, as shown in equation (6).

**Feathered Poisson seam fusion**

When the light difference and the colour difference between the reference image and the target image are large, the seam can still be found even if seamless stitching can be realized. Hence, Poisson fusion was used to process the stitched images. The binary mask of the seam is presented in Figure 5.
To provide more natural seam edges for subsequent fusion operations, the coordinates of any pixel on the binary mask of the seam were set as \((x_i, y_i)\). Among the pixels with a pixel value of zero, the pixel with the shortest quasi-Euclidean distance from \((x_i, y_i)\) was set as \((x_j, y_j)\). The binary mask of the seam is feathered according to the quasi-Euclidean distance:

\[
M(x_i, y_i) = \begin{cases} 
|x_i - x_j| + (\sqrt{2} - 1)|y_i - y_j|, & \text{if } |x_i - x_j| > |y_i - y_j| \\
(\sqrt{2} - 1)|x_i - x_j| + |y_i - y_j|, & \text{otherwise}
\end{cases}
\]

(8)

where \(M(x_i, y_i)\) is the new value of the pixel \((x_i, y_i)\) on the seam mask. By equation (8), we can obtain a new feathered seam mask, and the target image and reference image after applying the feathered seam.

Next, we discuss how to fuse the target and reference images after applying the feathered seam. In particular, the target image after seam operation is inserted in the reference image. The result needs to meet two requirements: it should be smooth to the extent possible, and the pixel values on the seam of the reference image and the target image should be consistent. The interpolation problem is shown in Figure 6.

\(T\) is the target image after seam-cutting, \(v\) is the gradient field corresponding to \(T\), \(R\) is the reference image after seam-cutting, and \(\Omega\) is a closed subset of \(R\) with boundary \(\partial \Omega\). To meet the two requirements mentioned above, the first requirement needs to meet the same gradient of \(T\) and \(R\) to ensure smoothness. Let the fused image pixels originating from \(\partial \Omega\) be represented by \(f\), and the rest be represented by \(f^*\). Then, the two requirements can be expressed as

\[
\begin{align*}
\min_f & \int_{\Omega} |\nabla f - v|^2 \\
\text{subject to } & f|_{\partial \Omega} = f^*|_{\partial \Omega}
\end{align*}
\]

(9)

where \(\nabla\) is the gradient operator, and the unique solution of equation (9) can be obtained by a Poisson equation with Dirichlet boundary conditions. The final fusion result can be obtained by solving the Poisson equation for the three colour components of the image.

The fusion results are presented in Figure 7. The left figure shows the results of the direct Poisson fusion, and the right figure shows the results of the feathered Poisson fusion with distance transformation. Through comparison, it is easy to find the seam at the ceiling directly using Poisson fusion, and there is unnatural exposure at the lampshade, whereas the feathering Poisson fusion with distance transformation is more natural in the transition effect.

**Algorithm**

The framework of semantic segmentation guided feature point classification and seam fusion for image stitching is shown in Algorithm 1.
Experiment

In this section, we compare the proposed algorithm with other mainstream software and algorithms on the ‘Interiornet’ dataset. The comparison algorithms include AUTOSTITCH, APAP, adaptive as-natural-as-possible (AANAP), shape-preserving-half-projective (SPHP), and robust-elastic-warping (REW).

Dataset

For the training of semantic segmentation networks, the training dataset was from the Massachusetts Institute of Technology’s ADE20K dataset, and the learning rate was set to 0.02. The training set contained 20210 images, the verification set had 2000 images, and the dataset contained various semantic scenes with 150 classes.

The ‘Interiornet’ dataset was used for image stitching, containing more than 20 types of scenes. The data of each scene were continuously collected according to the trajectory of the real camera and provided the variability of lighting and object sorting, which can provide challenging tasks for image stitching.

Results

In this section, the feasibility of the algorithm was proved by comparing the results of the proposed image stitching algorithm with other algorithms.

For scene 1, owing to the complexity of parallax and feature points, the AUTOSTITCH algorithm cannot obtain stitching results. As shown in Figure 8, from the overall performance of the stitching results, SPHP maintains a high degree of naturalness. Owing to the difference in exposure, APAP and AANAP can observe the overlapping area of the image. The three columns of the partial views on the right side of Figure 8 are from the yellow, blue, and red boxes in the stitching result diagram. Through observation, we found that different algorithms produce different degrees of ‘shaking’ at the ‘chair’ and ‘chandelier’. APAP produces perspective distortion, and AANAP and SPHP use similarity transformation to alleviate the perspective distortion of non-overlapping areas. Similarly, the chairs in the REW result are artefacts and the lights are slightly blurred. However, the proposed image stitching method addresses this problem by improving the alignment performance of the object classes.

For scene 2, there were significantly few feature points to be further grouped. Therefore, AANAP cannot calculate the rotation angle of the similarity transformation, and thus cannot produce a result. As shown in Figure 9, AUTOSTITCH sacrificed the shape for perfect alignment, resulting in the problem of ceiling bending; by observing

Algorithm 1 Semantic segmentation guided feature point classification and seam fusion for image stitching.

Input: The target image and the reference image.

Output: A stitching result.

1: The semantic segmentation results of target image and reference image are obtained through semantic segmentation network;
2: Detect, match and screen the feature point pairs of target image and reference image using the SIFT and RANSAC algorithm;
3: Classify the feature point pairs according to semantic segmentation results;
4: Calculate homography transformation \( h_j \) of each class of feature point through Eq. (4);
5: Calculate transformation error \( \varepsilon_h \) for all object class feature points in Eq. (5);
6: Calculate salient weight \( \delta(p, q) \) in Eq. (7) according to semantic segmentation results;
7: Calculate the semantic segmentation perception-based seam \( E \) in Eq. (6);
8: Calculate feathered seam mask in Eq. (8);
9: Fusion the target image and the reference image in Eq. (9).
the ‘tea table’, it can be found that AUTOSTITCH uses multi-band fusion to obtain clearer results, APAP and SPHP are blurred and repeated. The proposed algorithm obtains clearer results through object classes alignment and optimized Poisson fusion. By observing the ‘curtain’, we realize that the right side of APAP and SPHP curtain is inclined, and there are even two machines in the results of APAP in the last column. By examining the first two columns of detail, we observe that REW preserves the local shape well, but distorts the ceiling line.

The average gradient of the image represents the average value of the grey transformation rate of the image, which is commonly used to measure the rate of detail contrast.

Figure 8. Comparison with advanced image stitching methods on ‘Interiornet’ image.
Figure 9. Comparison with advanced image stitching methods on "Interiornet" image.
transformation of the image and can be used to represent the relative clarity of the image. The larger the value, the clearer the image, that is

\[ G = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \frac{(\partial f / \partial x)^2 + (\partial f / \partial y)^2}{2} \]

(10)

where \( x, y \) represent the abscissa and ordinate of the result, \( H, W \) represent the length and width of the result, and \( \partial f / \partial x, \partial f / \partial y \) represent the horizontal and vertical gradients, respectively.

To further verify the performance of the semantic segmentation guided feature point classification and seam fusion for image stitching, we calculated the average gradient of the results generated by a group of reference images and target images from the Interiornet dataset under different algorithms. A comparison of the results is shown in Table 1.

|                  | AUTOSTITCH | APAP | AANAP | SPHP | REW | OURS |
|------------------|------------|------|-------|------|-----|------|
| AVEGRAD          | 0.0269     | 0.0175 | 0.0160 | 0.0167 | 0.0216 | 0.0288 |

Table 1. Comparison of the quantitative performance of different approaches.

AVEGRAD: average gradient; APAP: as-projective-as-possible; AANAP: adaptive as-natural-as-possible; SPHP: shape-preserving-half-projective; REW: robust-elastic-warping.

Declaration of conflicting interests
The authors declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding
The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (61771141, 62172098, and 11771084), the Natural Science Foundation of Fujian Province (2020J01497), and the Engineering Research Center for ICH Digitalization and Multi-source Information Fusion (Fujian Polytechnic Normal University) under grant no. G3-KF1905.

ORCID iDs
Huafeng Huang https://orcid.org/0000-0003-2407-6977
Fei Chen https://orcid.org/0000-0002-3676-6011

References
1. Szeliski R. Image alignment stitching: A tutorial. Found Trends Comput Graph Vis 2006; 2(1): 1–104.
2. Fischler MA and Bolles RC. Random sample consensus a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 1981; 24(6): 381–395.
3. Brown M and Lowe DG. Automatic panoramic image stitching using invariant features. Int J Comput Vis 2007; 74(1): 59–73.
4. Gao J, Kim SJ and Brown MS. Constructing image panoramas using dual-homography warping. In: IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, June 21–23, 2011, pp. 49–56. IEEE.
5. Zaragoza J, Chin TJ, Brown MS, et al. As-projective-as-possible image stitching with moving DLT. In: IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 25–27, 2013, pp. 2339–2346. IEEE.
6. Lin WY, Liu S, Matsushita Y, et al. Smoothly varying affine stitching. In: IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, June 21–23, 2011, pp. 345–352. IEEE.
7. Zhang F and Liu F. Parallax-tolerant image stitching. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 24–27, 2014, pp. 3262–3269. IEEE.
8. Bakar SA, Jiang X, Gui X, et al. Image stitching for chest digital radiography using the sift and surf feature extraction by RANSAC algorithm. Journal of Physics: Conference Series. 2020; 1624(4): 042023.
9. Chen Y, Zheng H, Ma Y, et al. Image stitching based on angle-consistent warping. *Pattern Recognit* 2021; 117(4): 107993.
10. Jung K and Hong J. Quantitative assessment method of image stitching performance based on estimation of planar parallax. *IEEE Access* 2021; 9(99): 6152–6163.
11. Wang Z and Yang Z. Review on image-stitching techniques. *Multimedia Syst* 2020; 26(4): 413–430.
12. Nie L, Lin C, Liao K, et al. Unsupervised deep image stitching: Reconstructing stitched features to images. *IEEE Trans Image Process* 2021; 30: 6184–6197.
13. Jia Q, Li Z, Fan X, et al. Leveraging line-point consistence to preserve structures for wide parallax image stitching. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12186–12195.
14. Lin CC, Pankanti SU, Natesan Ramamurthy K, et al. Adaptive as-natural-as-possible image stitching. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 8–10, 2015, pp. 1155–1163. IEEE.
15. Chang CH, Sato Y and Chuang YY. Shape-preserving half-projective warps for image stitching. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 24–27, 2014, pp. 3254–3261. IEEE.
16. Li J, Wang Z, Lai S, et al. Parallax-tolerant image stitching based on robust elastic warping. *IEEE Trans Multimedia* 2017; 20(7): 1672–1687.
17. Lin K, Jiang N, Cheong LF, et al. Seagull: Seam-guided local alignment for parallax-tolerant image stitching. In: *European conference on computer vision*, Amsterdam, The Netherlands, October 8–16, 2016, pp. 370–385. Springer.
18. Boykov Y, Veksler O and Zabih R. Fast approximate energy minimization via graph cuts. *IEEE* 2001; 23: 1222–1239.
19. Boykov Y and Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans Pattern Anal Mach Intell* 2004; 26(9): 1124–1137.
20. Kolmogorov V and Zabin R. What energy functions can be minimized via graph cuts?. *IEEE Trans Pattern Anal Mach Intell* 2004; 26(2): 147–159.
21. Kwatra V, Schödl A, Essa I, et al. Graphcut textures: Image and video synthesis using graph cuts. *ACM Trans Graph* 2003; 22(3): 277–286.
22. Levin A, Zomet A, Peleg S, et al. Seamless image stitching in the gradient domain. In: *European conference on computer vision*, Zurich, Switzerland, September 6–12, 2004, pp. 377–389. Springer.
23. Yu T, Song K, Miao P, et al. Nighttime single image dehazing via pixel-wise alpha blending. *IEEE Access* 2019; 7: 114619.
24. Gao J, Li Y, Chin TJ, et al. Proceedings of Eurographics 2013. In The 34th Annual Conference of the European Association for Computer Graphics, Girona, Spain: The Eurographics Association, 2013, pp. 45–48. IEEE.
25. Li N, Liao T and Wang C. Perception-based seam cutting for image stitching. *Signal Image Video Process* 2018; 12(5): 967–974.
26. Pérez P, Gangnet M and Blake A. Poisson image editing. *ACM Transactions on Graphics (TOG)*, 2003, pp. 313–318.
27. Dai J, He K and Sun J. Convolutional feature masking for joint object and stuff segmentation. 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, June 8–10, 2015, pp. 3992–4000.
28. Dai J, He K and Sun J. Instance-aware semantic segmentation via multi-task network cascades. Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, USA, June 26th–July 1st, 2016, pp. 3150–3158. IEEE.
29. Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, Hawaii, July 21–26, 2017, pp. 2881–2890. IEEE.
30. Chen LC, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans Pattern Anal Mach Intell* 2017; 40(4): 834–848.
31. Long J, Shelhamer E and Darrell T. Fully convolutional networks for semantic segmentation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, pp. 3431–3440.
32. Yu F and Koltun V. Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122, 2016.
33. Ng PC and Henikoff S. Sift: Predicting amino acid changes that affect protein function. *Nucleic Acids Res* 2003; 31(13): 3812–3814.
34. Li W, Saeedi S, McCormac J, et al. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. arXiv preprint arXiv:1809.00716, 2018.
35. Zhou B, Zhao H, Puig X, et al. Scene parsing through ade20k dataset. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, Hawaii, July 21–26, 2017, pp. 633–641. IEEE.
36. Liu F, Gleichner M, Jin H, et al. Content-preserving warps for 3d video stabilization. *ACM Trans Graph* 2009; 28(3): 1–9.