Natural Image Stitching with the Global Similarity Prior

Yu-Sheng Chen(✉) and Yung-Yu Chuang

Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
{nothinglo,cyy}@cmlab.csie.ntu.edu.tw

Abstract. This paper proposes a method for stitching multiple images together so that the stitched image looks as natural as possible. Our method adopts the local warp model and guides the warping of each image with a grid mesh. An objective function is designed for specifying the desired characteristics of the warps. In addition to good alignment and minimal local distortion, we add a global similarity prior in the objective function. This prior constrains the warp of each image so that it resembles a similarity transformation as a whole. The selection of the similarity transformation is crucial to the naturalness of the results. We propose methods for selecting the proper scale and rotation for each image. The warps of all images are solved together for minimizing the distortion globally. A comprehensive evaluation shows that the proposed method consistently outperforms several state-of-the-art methods, including AutoStitch, APAP, SPHP and ANNAP.

Keywords: Image stitching · Panoramas · Image warping

1 Introduction

Image stitching is a process of combining multiple images into a larger image with a wider field of view [17]. Early methods focus on improving alignment accuracy for seamless stitching, such as finding global parametric warps to bring images into alignment. Global warps are robust but often not flexible enough. For addressing the model inadequacy of global warps and improving alignment quality, several local warp models have been proposed, such as the smoothly varying affine (SVA) warp [12] and the as-projective-as-possible (APAP) warp [20].

This work was supported by Ministry of Science and Technology (MOST) and MediaTek Inc. under grants MOST 104-2622-8-002-002 and MOST 104-2628-E-002-003-MY3.

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46454-1_12) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part V, LNCS 9909, pp. 186–201, 2016.
DOI: 10.1007/978-3-319-46454-1_12

These methods adopt multiple local parametric warps for better alignment accuracy. Projective (affine) regularization is used for smoothly extrapolating warps beyond the image overlap and resembling a global transformation as a whole. The stitched images are essentially single-perspective. Thus, they suffer from the problem of shape/area distortion and parts of the stitched image could be stretched severely and non-uniformly. The problem is even aggravated when stitching multiple images into a very wide angle of view. In such a case, the distortion accumulates and the images further away from the based image are often significantly stretched. Therefore, the field of view for the stitched image often has a limit. Cylindrical and spherical warps address the problem with a fairly narrow view of the perspective warp by projecting images onto a cylinder or a sphere. Unfortunately, these warps often curve straight lines and are only valid if all images are captured at the same camera center.

Recently, several methods attempt to address the issues with distortion and limited field of view in the stitched image while keeping good alignment quality. Since a single-perspective image with a wide field of view inevitably introduces severe shape/size distortion, these methods provide a multi-perspective stitched image. Chang et al. proposed the shape-preserving half-projective (SPHP) warp which is a spatial combination of a projective transformation and a similarity transformation [4]. SPHP smoothly extrapolates the projective transformation of the overlapping region into the similarity transformation of the non-overlapping region. The projective transformation maintains good alignment in the overlapping region while the similarity transformation of the non-overlapping region keeps the original perspective of the image and reduces distortion. In addition to projective transformations, SPHP can also be combined with APAP for better alignment quality. However, the SPHP warp has several problems. (1) The SPHP warp is formed by analyzing the homography between two images. It inherits the limitations of homography and suffers from the problem of a limited field of view. Thus, it often fails when stitching many images. (2) SPHP handles distortion better if the spatial relations among images are 1D. When the spatial relations are 2D, SPHP could still suffer from distortions (Fig. 5 as an example). (3) As pointed out by Lin et al. [11], SPHP derives the similarity transformation from the homography. If using the global homography, the derived similarity transformation could exhibit unnatural rotation (Fig. 4(e) as an example). They proposed the adaptive as-natural-as-possible (AANAP) warp for addressing the problem with the unnatural rotation. The AANAP warp linearizes the homography and slowly changes it to the estimated global similarity transformation that represents the camera motion. AANAP still suffers from a couple of problems. First, there are still distortions locally when stitching multiple images (Figs. 4(f), 5 and 6). Second, the estimation of the global similarity transformation is not robust and there could still exist unnatural rotation and scaling (Figs. 1(b), 3 and 5).

We propose an image stitching method for addressing these problems and robustly synthesizing natural stitched images. Our method adopts the local warp model. The warping of each image is guided by a grid mesh. An objective

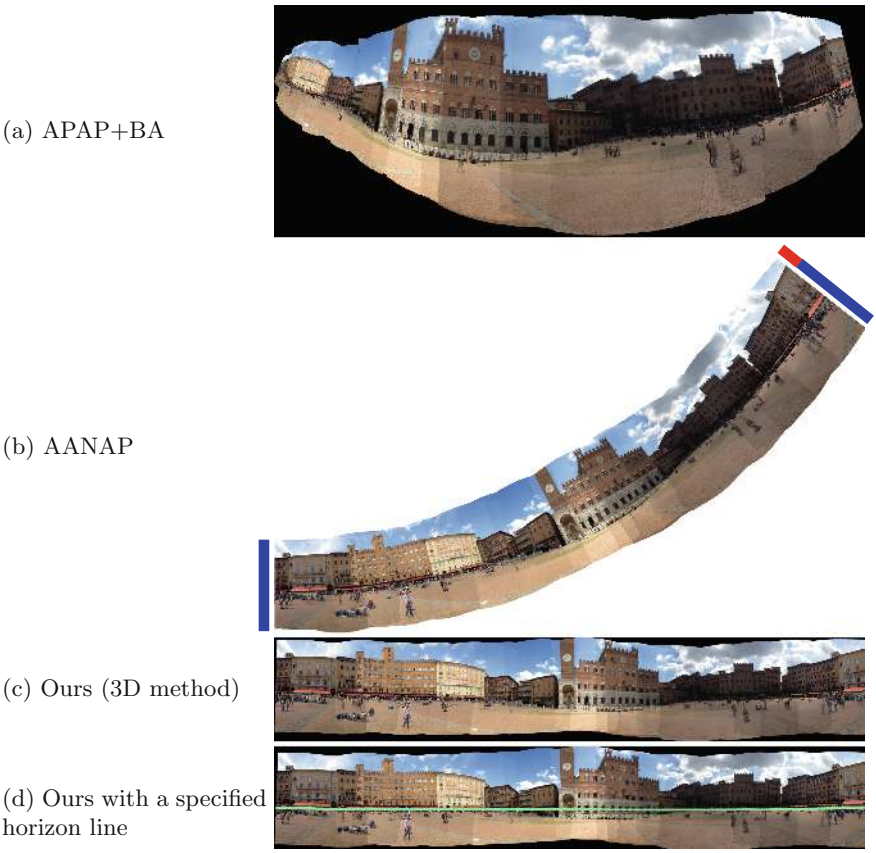

Fig. 1. Image stitching of 18 images.

function is designed for specifying the desired characteristics of the warps. The warps of all images are solved together for an optimal solution. The optimization leads to a sparse linear system and can be solved efficiently. The key idea is to add a global similarity term for requiring that the warp of each image resembles a similarity transformation as a whole. Previous methods have shown that similarity transformations are effective for reducing distortions [4,11], but they are often imposed locally. In contrast, we propose a global similarity prior for each image, in which proper selection of the scale and the rotation is crucial to the naturalness of the stitched image. From our observation, rotation selection is essential to the naturalness. Few paid attention to the rotation selection problem for image stitching. AutoStitch assumes that users rarely twist the camera relative to the horizon and can straighten wavy panoramas by computing the up vector [2]. AANAP uses feature matches for determining the best similarity transformation [11]. These heuristics are however not robust enough. We propose robust methods for selecting the proper scale and rotation for each image.

Our method has the following advantages. First, it does not have the problem with a limited field of view, a problem shared by APAP and SPHP. Second, by solving warps for all images together, our approach minimizes the distortion globally. Finally, it assigns the proper scale and rotation to each image so that the stitched image looks more natural than previous methods. In brief, our method achieves the following goals: accurate alignment, reduced shape distortion, naturalness and without a limit on the field of view. We evaluated the proposed method on 42 sets of images and the proposed method outperforms AutoStitch, APAP, SPHP and AANAP consistently. Figure 1 showcases common problems of previous methods. In Fig. 1(a), APAP+BA (Bundle Adjustment) [21] overcomes the problem with limited field of view by projecting images onto a cylinder. It however uses the wrong scale and rotation and the result exhibits non-uniform distortions over the image. AANAP does not select the rotations and scales properly. The errors accumulate and curve the stitching result significantly in Fig. 1(b). Our result (Fig. 1(c)) looks more natural as it selects the scales and the rotations properly. Our method can also incorporate horizon detection and the result can be further improved (Fig. 1(d)).

2 Related Work

Szeliski has a comprehensive survey on image stitching [17]. Image stitching techniques often utilize parametric transformations to align images either globally or locally. Early methods used global parametric warps, such as similarity, affine and projective transformations. Some assumed that camera motion contains only 3D rotations. A projection is performed to map the viewing sphere to an image plane for obtaining a 2D composite image. A noted example is the AutoStitch method proposed by Brown et al. [1]. Gao et al. proposed the dual-homography warping to specifically deal with scenes containing two dominant planes [5]. The warping function is defined by a linear combination of two homographies with spatially varying weights. Since their warp is based on projective transformations, the resulting image suffers from projective distortion (which stretches and enlarges regions).

Local warp models adopt multiple local parametric warps for better alignment accuracy. Lin et al. prioneered the local warp model for image stitching by using a smoothly varying affine stitching field [12]. Their warp is globally affine while allowing local deformations. Zaragoza et al. proposed the as-projective-as-possible warp which is globally projective while allowing local deviations for better alignment [20].

Instead of focusing on alignment quality, several methods address the problem with distortion in the stitched images. Chang et al. proposed the shape-preserving half-projective warp which is a spatial combination of a projective transformation and a similarity transformation [4]. The projective transformation maintains good alignment in the overlapping region while the similarity transformation of the non-overlapping region keeps the original perspective of

the image and reduces distortion. This approach could lead to unnatural rotations at times. Lin et al. proposed the adaptive as-natural-as-possible (AANAP) warp for addressing the problem with the unnatural rotation [11].

A few projection models have been proposed for reducing the induced visual distortion due to projection. Zelnik-Manor et al. used a multi-plane projection as an alternative to the cylindrical projection [22]. Kopf et al. proposed the locally adapted projection which is globally cylindrical while locally perspective [9]. Carroll et al. proposed the content-preserving projection for reducing distortions of wide-angle images [3]. When the underlying assumptions of these models are not met, misalignment occurs and post processing methods (e.g., deghosting and blending) can be used to hide it.

3 Method

Our method adopts the local warp model and consists of the following steps:

1. Feature detection and matching
2. Image match graph verification [2]
3. Matching point generation by APAP [20]
4. Focal length and 3D rotation estimation
5. Scale and rotation selection
6. Mesh optimization
7. Result synthesis by texture mapping

The input is a set of N images, $I_1, I_2, \ldots, I_N$. Without loss of generality, we use I_0 as the reference image. We first detect features and their matches in each image by SIFT [13]. Step 2 determines the adjacency between images. In terms of the quality of pairwise alignment, APAP performs the best. Thus, step 3 applies APAP for each pair of adjacent images and uses the alignment results for generating matching points. Details will be given in Sect. 3.1. Our method stitches images by mesh deformation. Section 3.2 describes our design of the energy function. To make the stitching as natural as possible, we add a global similarity term which requires each deformed image undergo a similarity transform. To determine the similarity transform for each image, our method estimates the focal length and 3D rotation for each image (step 4) and then selects the best scale and rotation (step 5). Section 4 describes the details of these two steps. Finally, the result is synthesize by steps 6 and 7.

3.1 Matching Point Generation by APAP

Let $\mathbf{J}$ denote the set of adjacent image pairs detected by step 2. For a pair of adjacent images I_i and I_j in $\mathbf{J}$, we apply APAP to align them using features and matches from step 1. Note that APAP is a mesh-based method and each image has a mesh for alignment. We collect I_i's mesh vertices in the overlap of I_i and I_j as the set of matching points, $\mathbf{M}^{ij}$. For each matching point in $\mathbf{M}^{ij}$, we know

(a) feature points (b) matching points (left to right)

Fig. 2. Feature points versus matching points. (a) feature points and their matches. (b) matching points and their matches.

its correspondence in I_j since I_i and I_j have been aligned by APAP. Similarly, we have a set of matching points $\mathbf{M}^{ji}$ for I_j.

Figure 2 gives an example of matching points. Given the features and matches in Fig. 2(a), we use APAP to align two images. After alignment, for the left image, we have a set of matching points which are simply the grid points in the overlap regions after APAP alignment. For these matching points, we have their correspondences in the right image. In further steps, we use matching points in place of feature points because matching points are distributed more uniformly.

3.2 Stitching by Mesh Deformation

Our stitching method is based on mesh-based image warping. For each image, we use a grid mesh to guide the image deformation. Let $\mathbf{V}_i$ and $\mathbf{E}_i$ denote the set of vertices and edges in the grid mesh for the image I_i. $\mathbf{V}$ denotes the set of all vertices. Our stitching algorithm attempts to find a set of deformed vertex positions $\tilde{\mathbf{V}}$ such that the energy function $\Psi(\mathbf{V})$ is minimized. The criteria for good stitching could be different from applications to applications. In our case, we stitch multiple images onto a global plane and would like to have the stitched image look as natural as the original images. About the definition of naturalness, we assume that the original images are natural to users. Thus, locally, our method preserves the original perspective of each image as much as possible. At the same time, globally, it attempts to maintain a good structure by finding proper scales and rotations for images. Both contributes to the naturalness of the stitching. Thus, our energy function consists of three terms: the alignment term Ψ_a, the local similarity term Ψ_l and the global similarity term Ψ_g.

Alignment term Ψ_a. This term ensures the alignment quality after deformation by keeping matching points aligned with their correspondences. It is defined as

$$\Psi_a(\mathbf{V}) = \sum_{i=1}^{N} \sum_{(i,j)\in\mathbf{J}} \sum_{p_k^{ij}\in\mathbf{M}^{ij}} \|\tilde{v}(p_k^{ij}) - \tilde{v}(\Phi(p_k^{ij}))\|^2, \tag{1}$$

where $\Phi(p)$ returns the correspondence for a given matching point p. The function $\tilde{v}(p)$ expresses p's position as a linear combination of four vertex positions, $\sum_{i=1}^{4} \alpha_i \tilde{v}_i$ where $\tilde{v}_i$ denote the four corners of the quad that p sits in and α_i are the corresponding bilinear weights.

Local similarity term Ψ_l. This term serves for regularization and propagates alignment constraints from the overlap regions to the non-overlap ones. Our choice for this term is to ensure that each quad undergoes a similarity transform so that the shape will not be distorted too much.

$$\Psi_l(\mathbf{V}) = \sum_{i=1}^{N} \sum_{(j,k)\in\mathbf{E}_i} \|(\tilde{v}_k^i - \tilde{v}_j^i) - \mathbf{S}_{jk}^i (v_k^i - v_j^i)\|^2, \tag{2}$$

where v_j^i is the position for an original vertex and $\tilde{v}_j^i$ represents the position of the vertex after deformation. $\mathbf{S}_{jk}^i$ is a similarity transformation for the edge (j,k) which can be represented as

$$\mathbf{S}_{jk}^i = \begin{bmatrix} c(e_{jk}^i) & s(e_{jk}^i) \\ -s(e_{jk}^i) & c(e_{jk}^i) \end{bmatrix}. \tag{3}$$

The coefficients $c(e_{jk}^i)$ and $s(e_{jk}^i)$ can be expressed as linear combinations of vertex variables. Details can be found in [8].

Global similarity term Ψ_g. This term requires each deformed image undergo a similarity transform as much as possible. It is essential to the naturalness of the stitched image. In brief, without this term, the results could be oblique and non-uniformly deformed as exhibited by AANAP and SPHP (Figs. 4 and 5). In addition, it eliminates the trivial solution, $v_j^i = 0$. The procedure for determining the proper scale and rotation is described in Sect. 4. Assume that we have determined the desired scale s_i and rotation angle θ_i for the image I_i. The global similarity term is defined as

$$\Psi_g(\mathbf{V}) = \sum_{i=1}^{N} \sum_{e_j^i\in\mathbf{E}_i} w(e_j^i)^2 \left[(c(e_j^i) - s_i\cos\theta_i)^2 + (s(e_j^i) - s_i\sin\theta_i)^2\right], \tag{4}$$

which requires the similarity transform for each edge e_j^i in I_i resembles the similarity transform we have determined for I_i. The functions $c(e)$ and $s(e)$ return the expressions for the coefficients of the input edge e's similarity transform as described in Eq. 3. The weight function $w(e_j^i)$ assigns more weight to the edges further away from the overlapped region. For quads in the overlap region, alignment plays a more important role. On the other hand, for edges away from the overlap region, the similarity prior is more important as there is no alignment constraint. Specifically, it is defined as

$$w(e_j^i) = \beta + \frac{\gamma}{|Q(e_j^i)|} \sum_{q_k\in Q(e_j^i)} \frac{d(q_k, \mathbf{M}^i)}{\sqrt{R_i^2 + C_i^2}}, \tag{5}$$

where β and γ are constants controlling the importance of the term; $Q(e^i_j)$ is the set of guads which share the edge e^i_j (1 or 2 quads depending on whether the edge is on the border of the mesh); $\mathbf{M}^i$ denotes the set of quads in the overlap region of I_i; the function $d(q_k, \mathbf{M}^i)$ returns the distance of the quad q_k to the quads in the overlap regions in the grid space; R_i and C_i denote the numbers of rows and columns in the grid mesh for I_i. At a high level, an edge's weight is proportional to the normalized distance of the edge to the overlap regions in the grid space.

The optimal deformation of meshes is determined by the following:

$$\tilde{\mathbf{V}} = \arg\min_{\tilde{\mathbf{V}}} \Psi_a(\mathbf{V}) + \lambda_l \Psi_l(\mathbf{V}) + \Psi_g(\mathbf{V}). \tag{6}$$

Note that there are two parameters, β and γ, in Ψ_g, controlling the relative importance of the global similarity term. In all of our experiments, we set $\lambda_l = 0.56$, $\beta = 6$ and $\gamma = 20$. Empirically, we found the parameters are quite stable because there is not severe conflict between terms. The optimization can be efficiently solved by a sparse linear solver.

4 Scale and Rotation Selection

This section describes how to determine the best scale s_i and rotation θ_i for each image I_i, which is the key to the naturalness of the stitched result.

4.1 Estimation of the Focal Length and 3D Rotation

We estimate the focal length and 3D rotation for each image by improving the bundle adjustment method proposed by AutoStitch [2]. We improve their method in two ways: better initialization and better point matches. Better initialization improves convergence of the method.

From a homography between two images, we can estimate the focal lengths of the two images [16–18]. After performing APAP, we have a homography for each quad of a mesh. Thus, each quad gives us an estimation of the focal length of the image. We take the median of these estimations as the initialization of the focal length and form the initial intrinsic matrix $\mathbf{K}_i$ for I_i. Once we have $\mathbf{K}_i$, we obtain the initial guess for 3D rotation $\mathbf{R}_{ij}$ between I_i and I_j by minimizing the following projection error:

$$\mathbf{R}_{ij} = \arg\min_{\mathbf{R}} \sum_{p^{ij}_k \in \mathbf{M}^{ij}} \|\mathbf{K}_j \mathbf{R} \mathbf{K}_i^{-1} p^{ij}_k - \Phi(p^{ij}_k)\|^2. \tag{7}$$

It can be solved by SVD. Note that AutoStich uses features and their matches for estimating the 3D rotation between two images. The problem with features is that they are not uniformly distributed in the image space and it could have adverse influence. We use matching points instead of feature points for estimating 3D rotation.

With the better initialization of $\mathbf{K}_i$ and $\mathbf{R}_{ij}$, bundle adjustment is performed for obtaining the focal length f_i and the 3D rotation $\mathbf{R}_i$ for each image I_i. The scale s_i for I_i in Eq. 4 can be set as

$$s_i = f_0 / f_i. \tag{8}$$

4.2 Rotation Selection

As mentioned in Sect. 1, although the selection of rotation is crucial to the naturalness, few paid attention to it. AutoStitch assumes that users rarely twist the camera relative to the horizon and can straighten wavy panoramas by computing the up vector [2]. AANAP uses feature matches for determining the best similarity transformation [11]. The heuristic is not robust enough as illustrated in Fig. 3.

(a) AANAP (b) Ours (3D method)

Fig. 3. AANAP does not select the right rotation (a). Our method does a better job and generates a more natural result.

The goal of rotation selection is to assign a rotation angle θ_i for each image I_i. We propose a couple of methods for determining the rotation, a 2D method and a 3D method. Before describing these methods, we define several terms first.

Relative rotation range. Given a pair of adjacent images I_i and I_j, each pair of their matching points uniquely determines a relative rotation. Assume that the k-th pair of matching points gives us the relative rotation angle θ_k^{ij}. We define the relative rotation range Θ^{ij} between I_i and I_j as

$$\Theta^{ij} = [\theta_{min}^{ij}, \theta_{max}^{ij}], \tag{9}$$

where $\theta_{min}^{ij} = \min_k \theta_k^{ij}$ and $\theta_{max}^{ij} = \max_k \theta_k^{ij}$.

Minimum Line Distortion Rotation (MLDR). Human is more sensitive to lines. Thus, we propose a procedure for finding the best relative rotation between two adjacent images with respect to line alignment. We first detect lines using

the LSD detector [6]. Through the alignment given by APAP, we can find the correspondences of lines between two adjacent images, I_i and I_j. Each pair of corresponding lines uniquely determines a relative rotation. We use RANSAC as a robust voting mechanism to determine the relative rotation between I_i and I_j. The voting power of each line depends on the product of its length and width. The final relative rotation is taken as the average of all inliers' rotation angles. We denote ϕ^{ij} as the relative rotation angle between I_i and I_j determined by MLDR.

Given all relative rotation angles ϕ^{ij} estimated by MLDR, we can find a set of rotation angles $\{\theta_i\}$ to satisfy the MLDR pairwise rotation relationship as much as possible. We represent θ_i as a unit 2D vector (u_i, v_i) and formulate the following energy function:

$$\mathbf{E}_{MLDR} = \sum_{(i,j)\in\mathbf{J}} \left\| R(\phi^{ij}) \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \begin{bmatrix} u_j \\ v_j \end{bmatrix} \right\|^2, \tag{10}$$

where $R(\phi^{ij})$ is the 2D rotation matrix specified by ϕ^{ij}. By minimizing $\mathbf{E}_{MLDR}$, we find a set of rotation angles θ_i to satisfy the MLDR pairwise rotation constraints as much as possible. To avoid the trivial solution, we need at least one more constraint for solving Eq. 10. We propose two methods for obtaining the additional constraints.

Rotation selection (2D method). In this method, we make a similar assumption with Brown et al. [2] by assuming that users rarely twist the camera relative to the horizon. That is, we prefer that $\theta_i = 0°$ if possible. First, we need to determine the rotation angle for one image. Without loss of generality, let the angle of the reference image be 0°, i.e., $\theta_0 = 0°$. Once we have the rotation angle θ_i for some image I_i, we can determine the rotation range of the image I_j adjacent to I_i by $\Theta_j = \Theta^{ij} + \theta_i$. If 0° is within the range Θ_j, it means that zero rotation is a reasonable one and we should set $\theta_j = 0$. By propagating the rotation ranges using BFS along the adjacency graph, we can find a set of images with 0° rotation. The pseudo code of the detailed process is given in the supplementary material. Let $\mathbf{\Omega}$ be the set of images whose rotation angles equal 0°. We find θ_i by minimizing

$$\mathbf{E}_{MLDR} + \lambda_z \mathbf{E}_{ZERO}, \text{ where} \tag{11}$$

$$\mathbf{E}_{ZERO} = \sum_{i\in\mathbf{\Omega}} \left\| \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\|^2 \tag{12}$$

and $\lambda_z = 1000$ so that the images in $\mathbf{\Omega}$ are likely assigned zero rotation, i.e., keeping their original orientations.

Rotation selection (3D method). In this method, we utilize the 3D rotation matrix $\mathbf{R}_i$ estimated at the beginning of this section. We first decompose the 3D

rotation matrix $\mathbf{R}_i$ to obtain the rotation angle α_i with respect to the z axis. The relative rotation between two adjacent images I_i and I_j can be determined as $\alpha^{ij} = \alpha_j - \alpha_i$. If $\alpha^{ij} \in \Theta^{ij}$, it means the estimation is reasonable and can be used. Otherwise, we should use the relative rotation ϕ^{ij} by MLDR. Let $\mathbf{\Omega}$ be the set of pairs which use ϕ^{ij} and $\bar{\mathbf{\Omega}} = \mathbf{J} - \mathbf{\Omega}$ for others. The rotation angles are determined by minimizing

$$\sum_{(i,j)\in\mathbf{\Omega}} \left\| R(\phi^{ij}) \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \begin{bmatrix} u_j \\ v_j \end{bmatrix} \right\|^2 + \lambda_r \sum_{(i,j)\in\bar{\mathbf{\Omega}}} \left\| R(\alpha^{ij}) \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \begin{bmatrix} u_j \\ v_j \end{bmatrix} \right\|^2. \quad (13)$$

We set $\lambda_r = 10$ to give 3D rotation more weights.

5 Experiments and Results

We compare our methods (2D and 3D versions) with four methods, AutoStitch [2], APAP [20], SPHP [4] and AANAP [11]. The experiments were performed on a MacBook Pro with 2.8 GHz CPU and 16 GB RAM. SIFT features were extracted using VLFeat [19]. The grid size is 40×40 for mesh-based methods. We tested the six methods on 42 sets of images (3 from [11], 6 from [4], 4 from [20], 7 from [14], 3 from [5] and 19 collected by ourselves). All comparisons can be found in supplementary material. The numbers of images range from 2 to 35. The test sets collected by us are more challenging than existing ones. We will release all our code and data for facilitating further comparisons.[1] Not account for feature detection and matching, for the resolution of 800×600, our method takes 0.1 s for stitching two images (Fig. 4) and 8 s for 35 images (Fig. 6).

Figure 4 compares all methods on stitching two images. Figure 4(a) shows the result of AutoStitch. Note that there is obvious misalignment. Our method can be used to empower other methods with APAP's alignment capability. Figure 4(b) shows the result in which the misalignment has been largely removed. Although with good alignment quality, APAP suffers from the problem with perspective distortion (Fig. 4(c)). One could change APAP's perspective model to similarity model as ASAP which is similar to the method by Schaefer et al. [15]. Figure 4(d) shows the result of ASAP. Although similarity performs well on reducing distortion, it is not effective for good alignment (closeup). In addition, the results would exhibit artifacts with obliqueness and non-uniform deformation. SPHP has the problem with unnatural rotation (Fig. 4(e)). AANAP gives a reasonable result in this example (Fig. 4(f)), but the lines on the floor are slightly distorted as shown more clearly in the closeup. Our method has the best stitching quality in this example (Fig. 4(g)).

Figure 1 presents an example for obtaining a panorama by stitching 18 images. SPHP failed on this example because of its limited field of view. APAP+BA overcomes the problem by projecting images onto a cylinder [21].

[1] The project website: http://www.cmlab.csie.ntu.edu.tw/project/stitching-wGSP/.

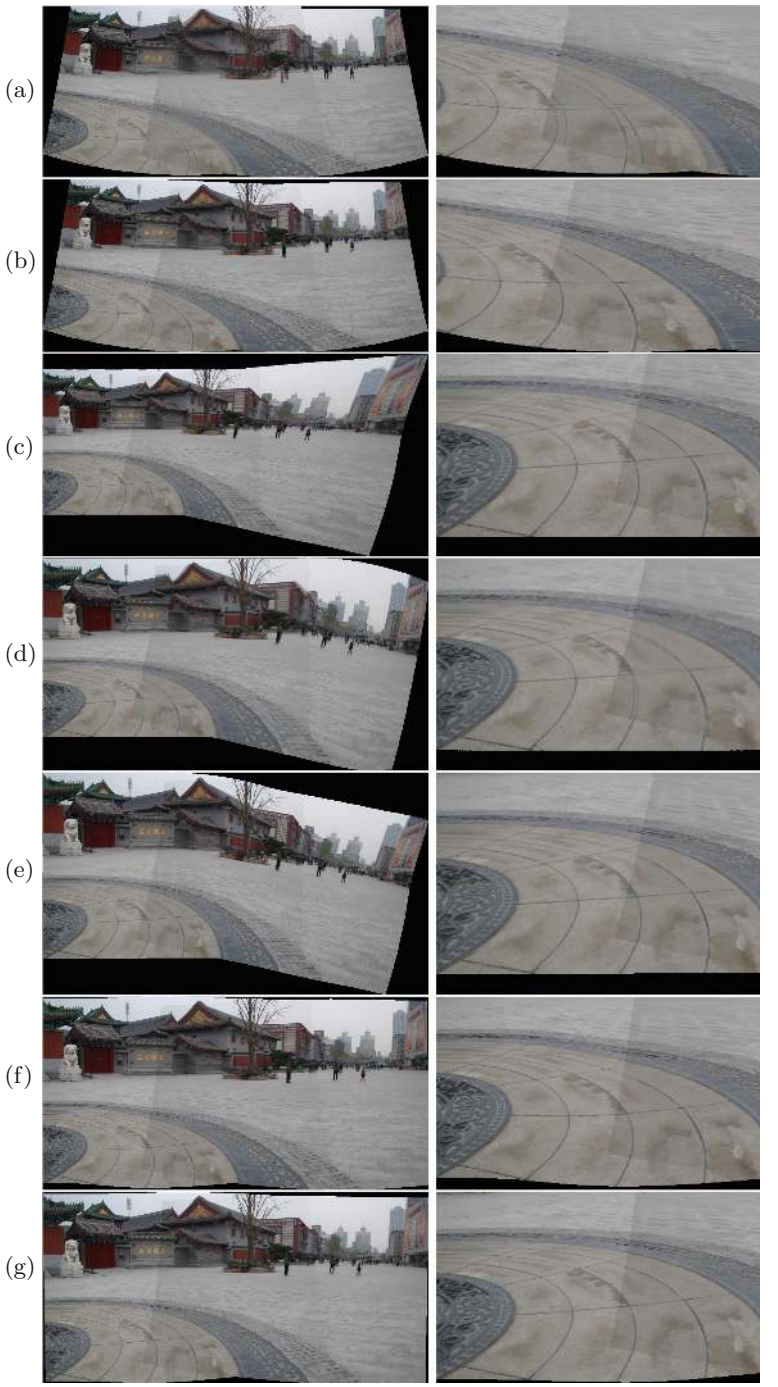

Fig. 4. An example of stitching two images. (a) AutoStitch, (b) AutoStitch+ours, (c) APAP, (d) ASAP, (e) SPHP+APAP, (f) AANAP, (g) Ours (3D method).

However, due to incorrect scale and rotation estimation, the result exhibits non-uniform distortions over the image (Fig. 1(a)). AANAP does not select the rotations and scales properly. The errors accumulate and curve the stitching result significantly as shown in Fig. 1(b). Note that the problem cannot be fixed by the rectangling panorama method [7] because it would maintain the original orientation of the input panorama as much as possible without referring to the original images. The panorama could become rectangular but the scene would remain curved. Our result (Fig. 1(c)) looks more natural as it selects the scales and the rotations properly. Our method is flexible and can be extended to comply with some additional constraints. In this example, we use a vanishing point detection method [10] for detecting the horizon for one image. With this additional constraint, the stitched image is better aligned with the horizon for a more natural result (Fig. 1(d)).

In the example of stitching six images in Fig. 5, AutoStitch introduces obvious distortion because of its spherical projection (top left). SPHP cannot handle 2D topology between images and suffers from distortion (bottom left). AANAP's result exhibits unnatural rotation and shape distortion (top right). Our result looks the most natural among all results (bottom right). The input of Fig. 6 contains 35 images. AutoStitch suffers from the distortion caused by the spherical projection (top left). AANAP has distortions all over the image (top right). Both of our methods give more natural results. The 2D method keeps the perspective

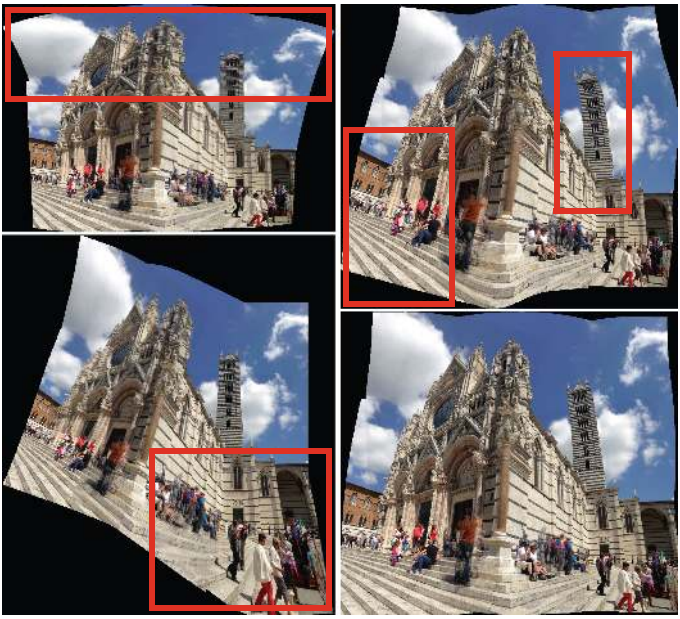

Fig. 5. An example of stitching six images. (top left) AutoStitch, (bottom left) SPHP+APAP, (top right) AANAP, (bottom right) Ours (2D method).

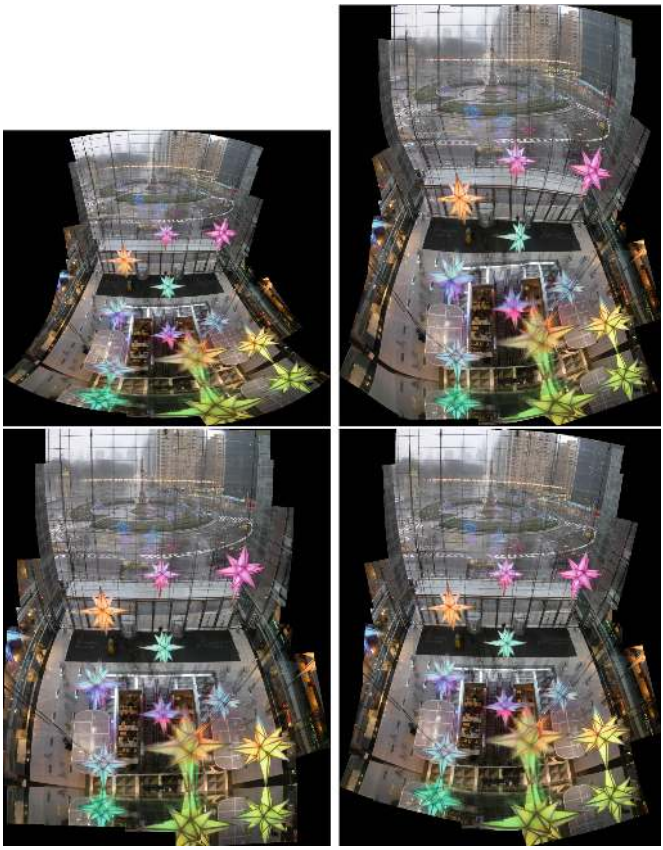

Fig. 6. An example of stitching 35 images. (top left) AutoStitch, (top right) AANAP, (bottom left) Our 2D method, (bottom right) Our 3D method.

of each image better (bottom left) while the 3D method keeps a better 3D perspective of the original scene (bottom right).

In sum, although ASAP, AANAP, SPHP and our method all use similarity, our method gives much better results. The differences come from how similarity is utilized. SPHP attempts to reduce the perspective distortion but it fails when the field of view is wide (Fig. 1) and the spatial relations among images are 2D (Fig. 5). AANAP attempts to address the unnatural rotation but it is not robust enough and fails frequently (Figs. 1(b), 3 and 5). In addition, AANAP does not optimize for shape distortion and it only stitches two images at a time. There could exist distortions locally when stitching multiple images (Figs. 4(f), 5 and 6). Our method addresses all these problems better than previous methods.

6 Conclusions

This paper proposes an image stitching method for synthesizing natural results. Our method adopts the local warp model. By adding the global similarity prior, our method can reduce distortion while keeping good alignment. More importantly, with our scale and rotation selection methods, the global similarity prior leads to a more natural stitched image.

This paper presents two main contributions. First, it presents a method for combining APAP's alignment accuracy and similarity's less distortion. Although individual components could have been explored, we utilize them in a different way. The method also naturally handles alignment of multiple images. Second, it presents methods for robustly estimating proper similarity transformations for images. They serve as two purposes: further enforcing similarity locally and imposing a good global structure. Experiments confirm the effectiveness and robustness of the proposed method.

References

1. Brown, M., Lowe, D.G.: Recognising panoramas. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, ICCV 2003, vol. 2, pp. 1218–1225 (2003)
2. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. Int. J. Comput. Vis. **74**(1), 59–73 (2007)
3. Carroll, R., Agrawal, M., Agarwala, A.: Optimizing content-preserving projections for wide-angle images. Int. J. Comput. Vis. **28**(3), 43 (2009)
4. Chang, C.H., Sato, Y., Chuang, Y.Y.: Shape-preserving half-projective warps for image stitching. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, pp. 3254–3261 (2014)
5. Gao, J., Kim, S.J., Brown, M.S.: Constructing image panoramas using dual-homography warping. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, pp. 49–56 (2011)
6. Grompone von Gioi, R., Jakubowicz, J., Morel, J.M., Randall, G.: LSD: a line segment detector. Image Process. On Line **2**, 35–55 (2012)
7. He, K., Chang, H., Sun, J.: Rectangling panoramic images via warping. ACM Trans. Graph. **32**(4), 79:1–79:10 (2013)
8. Igarashi, T., Igarashi, Y.: Implementing as-rigid-as-possible shape manipulation and surface flattening. J. Graph., GPU, & Game Tools **14**(1), 17–30 (2009)
9. Kopf, J., Lischinski, D., Deussen, O., Cohen-Or, D., Cohen, M.: Locally adapted projections to reduce panorama distortions. Int. J. Comput. Vis. **28**(4), 1083–1089 (2009)
10. Lezama, J., Grompone von Gioi, R., Randall, G., Morel, J.M.: Finding vanishing points via point alignments in image primal and dual domains. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014
11. Lin, C., Pankanti, S., Ramamurthy, K.N., Aravkin, A.Y.: Adaptive as-natural-as-possible image stitching. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, 7–12 June 2015, pp. 1155–1163 (2015)
12. Lin, W.Y., Liu, S., Matsushita, Y., Ng, T.T., Cheong, L.F.: Smoothly varying affine stitching. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, pp. 345–352 (2011)

13. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**, 91–110 (2004)
14. Nomura, Y., Zhang, L., Nayar, S.K.: Scene collages and flexible camera arrays. In: Proceedings of the 18th Eurographics Conference on Rendering Techniques, EGSR 2007, pp. 127–138 (2007)
15. Schaefer, S., McPhail, T., Warren, J.: Image deformation using moving least squares. In: ACM SIGGRAPH 2006 Papers, SIGGRAPH 2006, pp. 533–540 (2006)
16. Shum, H.Y., Szeliski, R.: Panoramic image mosaics. Technical Report MSR-TR-97-23, Microsoft Research, September
17. Szeliski, R.: Image alignment and stitching: a tutorial. Int. J. Comput. Vis. **2**(1), 1–104 (2006)
18. Szeliski, R., Shum, H.Y.: Creating full view panoramic image mosaics and environment maps. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997, pp. 251–258 (1997)
19. Vedaldi, A., Fulkerson, B.: Vlfeat: An open and portable library of computer vision algorithms. In: Proceedings of the 18th ACM International Conference on Multimedia, MM 2010, pp. 1469–1472 (2010)
20. Zaragoza, J., Chin, T.J., Brown, M.S., Suter, D.: As-projective-as-possible image stitching with moving DLT. In: Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2013, pp. 2339–2346 (2013)
21. Zaragoza, J., Chin, T.J., Tran, Q.H., Brown, M.S., Suter, D.: As-projective-as-possible image stitching with moving DLT. IEEE Trans. Pattern Anal. Mach. Intell. **36**(7), 1285–1298 (2014)
22. Zelnik-Manor, L., Peters, G., Perona, P.: Squaring the circle in panoramas. In: Proceedings of ICCV 2005, vol. 2, pp. 1292–1299 (2005)