SEAGULL: Seam-guided Local Alignment for Parallax-tolerant Image Stitching

Kaimo Lin[1,2] Nianjuan Jiang[2] Loong-Fah Cheong[1] Minh Do[2] Jiangbo Lu[2]

[1]National University of Singapore   [2]Advanced Digital Sciences Center, Singapore
linkaimo1990@gmail.com   Nianjuan.jiang@adsc.com.sg

Abstract. Image stitching with large parallax is a challenging problem. Global alignment usually introduces noticeable artifacts. A common strategy is to perform partial alignment to facilitate the search for a good seam for stitching. Different from existing approaches where the seam estimation process is performed sequentially after alignment, we explicitly use the estimated seam to guide the process of optimizing local alignment so that the seam quality gets improved over each iteration. Furthermore, a novel structure-preserving warping method is introduced to preserve salient curve and line structures during the warping. These measures substantially improve the effectiveness of our method in dealing with a wide range of challenging images with large parallax.

1 Introduction

Traditional image stitching techniques estimate a global 2D transformation (e.g. homography transformation) to align the input images [22, 3, 23]. The underlying assumption is that the images are taken at a fixed viewpoint or the scene is roughly planar. Violation of these assumptions will result in visual artifacts such as ghosting or misalignment that cannot be accounted for by a global 2D transformation. Such misalignment between the warped image and the reference image is referred to as parallax, and in this paper, we primarily want to address the problem of image stitching under large parallax.

For images with *small* parallax, some spatially-varying warping methods [25, 20, 19] combined with advanced image composition techniques like seam cutting [15, 2] and multi-band blending [4] usually suffice. However, when the images are taken from different viewpoints and the scene contains non-planar or discontinuous surfaces (often the case when the images are taken casually by users), most existing methods fail to produce satisfactory stitching results due to the presence of *large* parallax [26]. For images with large parallax, global alignment, as an over-simplified model to account for the underlying camera-scene geometry, cannot produce visually plausible stitching results (see Fig. 1). Instead, one only needs to find an alignment model that will produce good seams to stitch two images. The desiderata of a good seam is that it should either pass through non-salient homogeneous regions or salient regions if the latter are well-aligned

Fig. 1. Comparison of global alignment and our seam-guided local alignment. Left: Input images. Middle: Stitching result by APAP [25] (all features are used). Right: Stitching result by our method (only features around the final stitching seam are used).

locally. Therefore, the problem boils down to finding such a parallax-free local region for stitching. Recent works [11, 26] propose different strategies to select a subset of sparse feature matches that will facilitate finding such local regions for stitching. In these works, when the current alignment hypothesis is not satisfactory, a new set of features will be selected to generate an alternative alignment. This means the location of the current seam and its alignment quality are *not* used in any way to influence the new feature selection. Without exploiting the current results to decide or guide the next attempts, these existing methods have a few limitations: 1) the quality of a new seam might indeed be worse than the previous one; 2) if the scene in view is complex, then indeed one might have to generate a large number of alignment hypotheses before hitting upon a satisfactory one; and 3) it is non-trivial to decide the threshold setting that can be effectively used in all images for terminating the hypothesis generation process.

In this paper, we propose a *seam-guided local alignment (SEAGULL)* scheme for image stitching in the presence of large parallax. As its name suggests, we iteratively look for a good local alignment by performing seam-guided feature reweighting. Specifically, we weight the feature matches according to their current alignment errors (*i.e.*, the distance between two matching features after alignment) and their distances to the current estimated seam. This scheme stems from our observation that treating all the feature matches uniformly is usually not desirable in the presence of large parallax. For instance, methods [20, 25] that aim at global alignment across the entire overlapping region often suffer from noticeable local distortions due to parallax or misalignment at the estimated seam. It follows that feature matches with large misalignment or far away from the current estimated seam should be weighted down when computing the image alignment refinement (see Fig. 1 Right). Another motivation of our iterative alignment refinement scheme is that the current alignment does provide useful information to guide the search for a better seam. Generally, at least some parts of the estimated seam pass through well-aligned parallax-free regions. We stand a much better chance to obtain an improved seam by locally perturbing the current seam rather than trying out an entirely new one. This is a much more effective strategy to deal with scenes with large and complex parallax.

To overcome the local minima problem of the iterative seam refinement, we generate multiple initial alignment hypotheses from subsets of feature matches obtained by a *superpixel-based feature grouping* method. Each alignment hypothesis is then further refined by our seam-guided local alignment process. The optimized local alignment with the best stitching quality will be selected as the final stitching result. Our feature grouping method usually generates a small set of alignment hypotheses for optimization, yet because of the refinement process, these hypotheses are usually sufficient for obtaining a good stitching result.

The second contribution of our paper stems from the following observation. Many image alignment methods based on a subset of sparse feature matches do not have adequate control of the warping in image regions containing few features selected for alignment estimation. This results in noticeable distortion of the salient scene structures (e.g. lines and curves) in those regions. Even for regions with selected features, if these features contain certain amount of parallax, the warping may still suffer from unpleasant distortion in the aforementioned salient structures. Thus we propose a novel *structure-preserving warping* method that can effectively preserve curve and line structures during warping. We augment the basic CPW [20] framework with a new non-local structure-preserving term, so that similarity transformation constraints are enforced on the detected curve and line structures in the image, as well as on local mesh grids. Unlike the approach of [14, 5, 13, 7], our non-local structure-preserving term introduces a sparse linear system and can be easily integrated into many other mesh-based warping methods [20, 21]. [18] also introduces a line-preserving term in video stitching task. However, ours is more general and can also preserve curve structures.

2   Related Work

Image stitching is a well studied topic, yet stitching images with large parallax is still fraught with difficulties. A comprehensive survey can be found in [22]. Here, we briefly review related works from different perspectives.

Homography-based methods. Early methods [23, 3] employ only one single homography to align two images. These methods can generate good stitching results if the images are taken from the same viewpoint or the scenes are roughly planar. However, these assumptions can be easily violated in practice when parallax exists. Although advanced composition methods (e.g. multi-band blending [4], seam cut [15]) can be used to alleviate the problem to some extent, artifacts still remain especially when parallax is large. Gao *et al.* [10] used a dual-homography model to stitch images and obtained good results when the scene can be roughly modeled by two dominant planes.

Spatially-varying warps. Spatially-varying warping methods [20, 19, 25] are introduced to handle images with parallax. Combined with advanced composition techniques, these methods can be very effective for generating visually plausible stitching results for images with small parallax. In particular, Lin *et al.* [19]

estimated a smoothly varying affine field to align the images. Zaragoza *et al.* [25] proposed to use as-projective-as-possible warps to interpolate a smoothly varying projective stitching field. Li *et al.* [16] developed a dual-feature warping model for image alignment, using both the sparse feature matches and line correspondences. However, their method needs to predetermine line correspondences, which is a difficult task for images with large parallax. Also, for line structures without correspondences, this method cannot guarantee their straightness after warping. Our method, on the other hand, can preserve all curve and line structures effectively as long as they are detected.

Shape-preserving methods. Shape preserving warping methods mainly aim at generating natural-looking stitching results given a particular alignment model. Chang *et al.* [6] proposed a Shape-Preserving Half-Perspective (SPHP) warp that can smoothly transit from a projective transformation in the overlapping region into a similarity transformation in the non-overlapping region, with the latter aiming to counteract the unnatural perspective arising from the strange viewpoint (e.g. excessive tilting) associated with the projective transformation. Lin *et al.* [17] also proposed a warping model that combines two stitching fields (homography and global similarity) to generate natural-looking panoramas. These methods do not explicitly handle parallax.

Seam-driven stitching methods. While the existing spatially-varying methods have been demonstrated to work well on images with moderate parallax, Zhang and Liu [26] argued that they may fail on images with large parallax. Gao *et al.* [11] posited that there is no need to employ all the feature matches in estimating the warping model, so long as the ultimate objective is to generate visually plausible results. Zhang and Liu [26] used this observation and proposed a hybrid transformation model to handle images with large parallax. They combined homography warp and content-preserving warp (CPW) [20] to align images. A randomized feature selection algorithm is developed to hypothesize homography candidates that may lead to good stitching seams. As its name suggests, warping hypotheses are searched for in a randomized fashion, in which the current pass does not use the alignment knowledge gained from the previous iterations. Our method, on the other hand, iteratively refines the warping model by adjusting feature weights according to their distances to a particular seam.

3 Stitching Algorithm

In this section, we will first briefly introduce our stitching pipeline before introducing the details of each step. For clarity of exposition, we take the two-image stitching case as an example. We keep the reference image fixed and warp the target image. Stitching multiple images can be easily extended from this pipeline by adding one image at a time. As shown in Fig. 2, our stitching pipeline takes in multiple local alignment hypotheses as input, and applies seam-guided local alignment on each of these hypotheses to obtain locally optimal stitching. The final stitching is selected as the one with the best stitching seam quality.
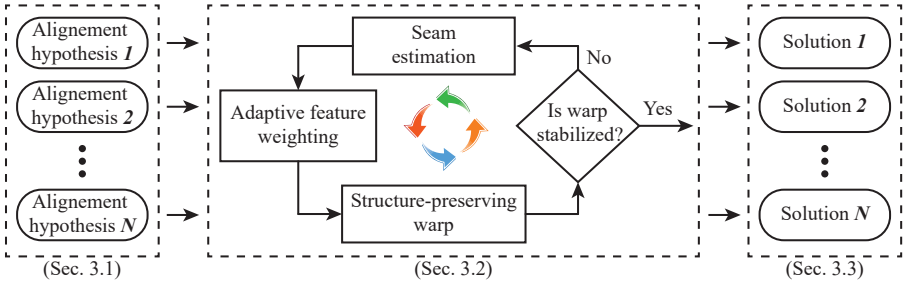
Fig. 2. The pipeline of our stitching algorithm.

3.1   Alignment Hypotheses Generation

For images with large parallax, it is shown that finding a local alignment that facilitates a seamless stitch is more effective in practice [11, 26]. In this paper, we propose multiple alignment hypotheses to locate a good seam for stitching. Our goal is to generate a small set of hypotheses that are representative and distinctive from each other. To that end, we use a superpixel-based feature grouping method. Specifically, we first use SIFT [24] to obtain an initial set of feature matches. Then, we over-segment the target image using the method in [1]. Our goal is to partition the superpixels that contain features into several representative superpixel groups. Before the grouping, we first remove those outlier features in each superpixel by performing homography fitting with RANSAC [9].

In the grouping process, only superpixels that contain features are used. At the very beginning, all the superpixels are labeled as 'ungrouped'. In each iteration, we initialize a superpixel group S_i with the ungrouped superpixel that has the largest number of features. Then, we check all the neighboring ungrouped superpixels of S_i and add them to the group one by one if the homography fitting error of the new group is less than 5 pixels[1]. The growing process terminates when no more neighboring superpixels of S_i can be added. We repeat the above process in the remaining ungrouped superpixels until all the superpixels have been assigned to a group. Given these superpixel groups, we also perform a merging step to further reduce the group number. Specifically, we merge two superpixel groups if the homography fitting error of the newly merged group is also smaller than 5 pixels. This merging step starts from the group with the largest number of features, and tries to merge all the other groups to it in descending order of group size. We repeat this process for the remaining unmerged groups until no more groups can be merged. Finally, we use features in each resulting superpixel group to estimate a local homography. Each warped target image is regarded as a local alignment hypthesis.

To avoid only generating local homography hypotheses which could be biased, we further enrich the hypothesis set by combining different superpixel groups to

---

[1] Unless otherwise noted, all the constant parameters in our algorithm are set for an image resolution of around 720p.

produce extra alignment hypotheses. The total number of such combinations is given by $C_k^2 + C_k^3 + ... + C_k^k$, where k is the number of the groups (usually $k = 1 \sim 4$).

3.2   Seam-guided Local Alignment

Our seam-guided local alignment optimizes each alignment hypothesis by iterating over the following three steps. Firstly, feature matches are weighted according to their current alignment errors and distances to the current estimated seam. Then, the target image is warped by a novel structure-preserving warping method. Finally, a stitching seam is estimated based on 'colored edge images'. The iteration terminates when there is little change of the mesh vertice locations compared to the previous iteration (average change less than one pixel) or the iteration number exceeds 5. For a reasonably good alignment hypothesis, this process usually terminates in $2 \sim 3$ iterations. Otherwise, we will just terminate those bad cases early by setting the hard limit of 5 iterations for run-time efficiency. Upon iteration termination, the final stitching seam quality is recorded.

Adaptive Feature Weighting In each iteration, we compute a weight for each feature match using the following expression:

$$w = \lambda \Big(e^{-\frac{d_m^2}{2\sigma_m^2}} + \epsilon \Big), \tag{1}$$

where the terms in the bracket depend on the current alignment error of the feature and λ depends on the distance of the feature to the current seam. Specifically, d_m is the distance between the feature in the warped target image and its correspondence in the reference image. The terms $\epsilon = 0.01$ and $\sigma_m = 10$ are constants. λ is set to 1.5 if $d_s \leq 20$ (d_s is the shortest distance from the feature to the current seam), and 0.1 otherwise. In the first iteration when the seam has not been estimated, all d_s are set to zero.

Structure-preserving Warp We use a $m \times n$ grid mesh to represent the target image. Image warping is achieved by texture mapping using the coordinates of mesh vertices after deformation. Our proposed structure-preserving warp consists of a feature term and two structure-preserving terms. Different from CPW [20], our structure-preserving terms include both local and non-local similarity constraints. The total energy function is given by the following:

$$E(\hat{V}) = \lambda_1 E_f(\hat{V}) + \lambda_2 E_{ls}(\hat{V}) + \lambda_3 E_{cs}(\hat{V}), \tag{2}$$

where $\hat{V}$ are the unknown coordinates of mesh vertices to be estimated. The feature term, local and non-local structure terms are denoted by $E_f(\hat{V})$, $E_{ls}(\hat{V})$, and $E_{cs}(\hat{V})$ respectively. The constant λ_1, λ_2, and λ_3 are the associated weights for these three terms ($\lambda_1 = 5$, $\lambda_2 = 1$, and $\lambda_3 = 10$ in our implementation). All these terms will form a sparse linear system which can be easily minimized.
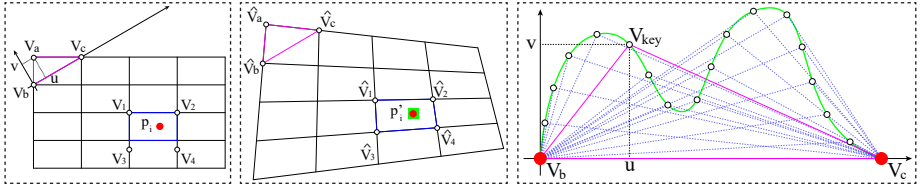
Fig. 3. Structure-preserving warp. Please refer to the text for details.

Feature term. The feature term is defined the same way as the data term in CPW [20]. As shown in Fig. 3 left, each feature point p_i can be represented by the 2D bilinear interpolation of the four vertices $(V_k, k = 1, ..., 4)$ of its enclosing grid cell. To align p_i to its matched location p'_i (green square in Fig. 3 middle) after deformation, we define the feature term as:

$$E_f(\hat{V}) = \sum_i w_i \| \sum_{k=1}^4 c_k \hat{V}_k - p'_i \|^2, \tag{3}$$

where $\hat{V}$ contains the unknown mesh vertices. The bilinear coefficients ($c_k, k = 1, ..., 4$) are used to determine the location of p_i after warping. The feature weight w_i for p_i will be updated during iterations as given in Eq. 1.

Structure-preserving terms. Our structure-preserving terms are defined on both local and non-local similarity transformation constraints. According to [20], in a triangle consisting of three vertices, the coordinates (u, v) for a vertex V_a in the local coordinate system defined by the other two vertices V_b and V_c is given by

$$V_a = V_b + u(V_c - V_b) + vR_{90}(V_c - V_b), R_{90} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{4}$$

For a triangle that undergoes a similarity transformation, the new vertex $\hat{V}_a$ can still be represented by $\hat{V}_b$ and $\hat{V}_c$ using the same local coordinates (u, v) computed from its initial shape. Hence, one can minimize the following cost to encourage similarity transformation on a given triangle,

$$C_{tri} = \| \hat{V}_a - (\hat{V}_b + u(\hat{V}_c - \hat{V}_b) + vR_{90}(\hat{V}_c - \hat{V}_b)) \|^2. \tag{5}$$

Locally, as shown in Fig. 3 left, each grid cell can be divided into two triangles. We sum up C_{tri} defined over all the triangles in the grid mesh to compute our local similarity term E_{ls}. This local similarity constraint is also used in [20] to maintain spatial smoothness of the warping. However, it does not provide sufficient constraints on salient structures larger than the size of the mesh cells. Therefore, we explicitly extract contours, and use triangles defined on each of the contours to compute a set of non-local similarity constraints. Specifically, we first extract contours from the target image using OpenCV's contour detection function. For contours with branching nodes, we break them at these nodes and

collect the sub-contours as curve segments. Otherwise, each contour is a curve segment. Curve segments with a length shorter than 20 pixels will be discarded. Then we uniformly sample key points (green curve in Fig. 3 right) along each curve and define a set of triangles formed by the two endpoints (red points in Fig. 3 right) and each key point. The non-local similarity term is thus given by

$$E_{cs}(\hat{V}) = \sum_{i=1}^{N_c} \sum_{j=1}^{N_k} \|\hat{V}_{key}^{i,j} - (\hat{V}_b^i + u(\hat{V}_c^i - \hat{V}_b^i) + vR_{90}(\hat{V}_c^i - \hat{V}_b^i))\|^2, \qquad (6)$$

where N_c is the total number of curve segments and N_k is the number of key points on each curve segment i. The curve vertices $\hat{V}_{key}^{i,j}$, $\hat{V}_b^i$, and $\hat{V}_c^i$ can be represented by the mesh vertices using bilinear interpolation just like the feature term. Note that the non-local structure-preserving term is also valid for line structures. Therefore, we also employ a line detector [12] to detect line segments in the target image and add them to the current curve set.

Seam Estimation To apply the seam cut technique [15], one first computes a difference map between the reference image and the target image in the over-lapping region. The difference map is usually obtained by calculating either the color difference of the pixels or the Canny edge map difference [26]. The pixel color difference approach has more discriminatory power compared to the Canny edge map approach, whereas the latter has stronger robustness against illumination changes. We combine the strength of both by retaining pixel colors that are near the extracted Canny edges, and refer to this representation as the 'colored edge image'. Specifically, we expand the edge map mask by 1 pixel on either side of the edge and retain the original color of the pixels on the expanded edge mask, with other pixels' colors set to black. Our stitching seam is obtained by applying the seam cut technique [15] on the 'colored edge images'.

3.3   Stitching Seam Quality Assessment

Since SEAGULL targets on local alignment and can preserve salient scene structures during the warp, we only need to evaluate the alignment quality along the final stitching seam. Specifically, for each pixel p_i on the final stitching seam, we first define a 15×15 local patch (in pixels) centered at p_i. Then we compute the ZNCC score between the local patch in the target image and that in the reference image. The seam quality is then defined as follows:

$$Q_{seam}(p) = \frac{1}{N} \sum_i^N (1.0 - \frac{ZNCC(p_i) + 1}{2}), \qquad (7)$$

where N is the total number of pixels on the seam, excluding the ones that are not on the colored edge masks.
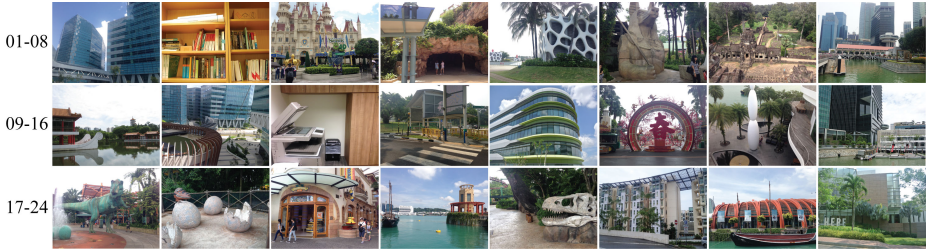
01-08

09-16

17-24

Fig. 4. Our dataset used in this paper.

4   Experiments

We demonstrate the effectiveness of SEAGULL in two aspects. Firstly, we conduct several experiments to validate the design of the individual components in SEAGULL, specifically, the alignment hypothesis generation, the seam-guided local alignment optimization, and the structure-preserving warping method. Secondly, we compare the overall performance of SEAGULL with two state-of-the-art stitching methods, APAP [25] and Zhang and Liu's method [26]. We evaluate the methods over two datasets: the first comprises of 24 pairs of images taken by us using mobile phones with challenging parallax variation (Fig. 4), and the second uses the images from Zhang and Liu's published dataset, which can be found on their project website. To suppress the intensity difference along the estimated seam, we apply the method from [8] to all the final stitching results.

4.1   Homography Hypothesis Evaluation

We compare the alignment hypothesis generation method in SEAGULL with that in [11] and [26], in which [11] is based on homography fitting with RANSAC [9], and [26] is based on randomized feature selection (for more details, refer to [26]). The experiment is conducted on our own dataset of 24 pairs of images. We use the same threshold for homography fitting errors in all three methods. For [26], the iteration terminates when the average penalty value of all the features is larger than a threshold. However, the value of the threshold is not reported in [26] and we find it quite tricky to set a universally appropriate value. If the value is too small, many features may not have the chance to be selected in the whole process. If the value is too large, each feature may be selected multiple times and the algorithm may generate many redundant homography hypotheses. In the presence of large and complex parallax, any feature may contribute to the search of a good stitching seam. Our goal is to try as many features as possible while keeping the number of hypotheses small. Therefore, we choose a different termination condition whereby the algorithm of [26] is terminated if more than 80% of the features have been selected at least once in the previous iterations.

Fig. 5 shows the comparison results. The top graph shows the number of alignment hypotheses generated by the respective methods. Since [26] contains
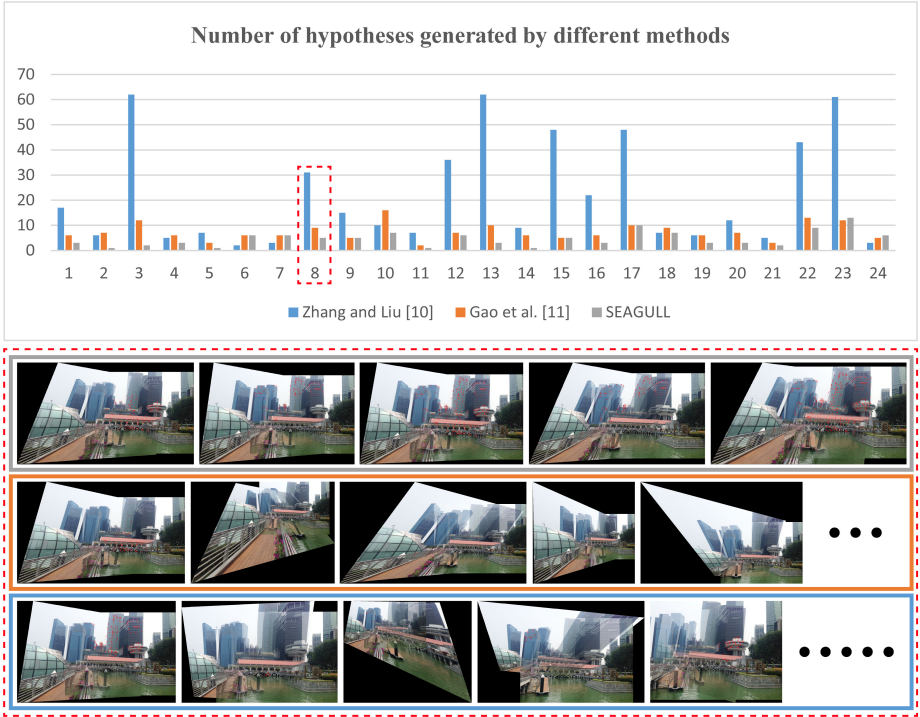
Fig. 5. Comparison of different hypothesis generation methods. The last three rows show example alignment hypotheses produced by SEAGULL, [11] and [26] respectively.

randomness in seed selection, we run the algorithm ten times and record the mean values. As can be seen from the graph, in most of the cases, SEAGULL generates the smallest set of alignment hypotheses. [26] usually generates more hypotheses than the other two methods. The reason is that its homography fitting process in each iteration will terminate immediately when one candidate feature can not be added to the current group regardless of the other unchecked nearest neighbors. This premature termination results in many small feature groups, given that there are inevitable feature mismatches. The bottom figure shows some alignment hypotheses generated by these methods on the image pair *No.8*. We can see that all of our results are fairly good for further optimization. However, some results from [11] and [26] are clearly unsuitable for stitching.

4.2   Seam-guided Local Optimization Evaluation

To demonstrate the effectiveness of our seam-guided local optimization, we compare the final stitching seam quality with and without the optimization part. The experiment is also conducted on our own dataset. For each example, we take the alignment hypothesis that leads to the best stitching result after local

Table 1. Stitching seam quality before and after seam-guided local optimization.

No.	Homo	Opti	Before	After	No.	Homo	Opti	Before	After
01.	1	1	0.186	**0.148**	13.	3	3	0.094	**0.045**
02.	1	1	**0.059**	0.061	14.	1	1	0.117	**0.074**
03.	**1**	**2**	0.227	**0.135**	15.	4	4	0.319	**0.205**
04.	2	2	0.326	**0.217**	16.	**3**	**1**	0.163	**0.138**
05.	1	1	**0.333**	0.387	17.	**1**	**4**	0.338	**0.114**
06.	**4**	**6**	0.150	**0.072**	18.	**4**	**1**	0.374	**0.336**
07.	**1**	**4**	0.275	**0.168**	19.	**1**	**3**	0.206	**0.142**
08.	**1**	**3**	0.166	**0.072**	20.	**3**	**1**	0.192	**0.170**
09.	**5**	**1**	0.189	**0.066**	21.	2	2	**0.164**	0.179
10.	4	4	0.265	**0.195**	22.	**1**	**3**	0.172	**0.080**
11.	1	1	0.288	**0.256**	23.	**8**	**10**	0.206	**0.159**
12.	6	6	0.273	**0.265**	24.	**4**	**1**	0.326	**0.148**
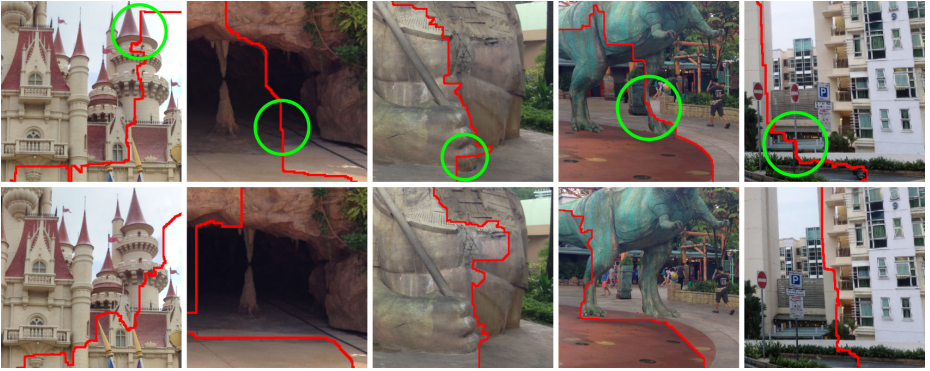
Fig. 6. Comparison of with and without our seam-guided local optimization. Top: Results without optimization. Bottom: Results with optimization.

optimization, and apply seam estimation on both alignments before and after the local optimization. We compare the respective seam quality in **Table 1**. Particularly, stitching seam quality with and without the local optimization is listed in columns **'After'** and **'Before'**, respectively. A smaller value usually indicates noticeable visually improved seam quality. As we can see, in most of the cases, the seam quality improves after our seam-guided local optimization. In some examples (i.e. 02, 05, and 21), the two stitching results share similar seam quality and are all visually plausible. Fig. 6 provides a visual comparison of stitching results with a difference in seam quality larger than 0.05. We can see that our seam-guided local optimization clearly improves the seam quality. Besides the seam quality comparison, we also record the index of the alignment hypothesis that leads to the best stitching with and without the optimization process. Column **'Homo'** in **Table 1** indicates the index of the alignment hy-
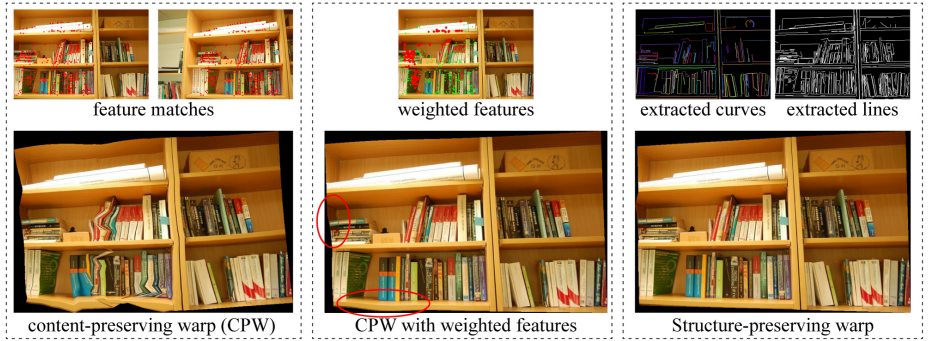
Fig. 7. Evaluations of our weighting scheme and non-local structure-preserving term. Left: CPW[20] with equally weighted features. Middle: CPW[20] with seam-guided weighted features. Right: Our structure-preserving warp.

pothesis that produces the best stitching result without our local optimization. Column **'Opti'** indicates the index of the best alignment hypothesis with the optimization. Interestingly, the homography hypothesis with the best stitching quality at the beginning does not always lead to the best stitching result after optimization.

4.3   Structure-preserving Warp Evaluation

Our structure-preserving warp effectively preserves salient curve and line structures during image warping while facilitating good local alignment around the estimated seam. An example using CPW [20] with equally weighted feature matches is shown in Fig. 7 left. Since the detected feature matches may contain wrong pairs and the parallax is too large for 2D global alignment, salient curve and line structures are severely distorted during the warp. Using global homography fitting to remove the mismatches is not a good practice for images with large parallax, since it may also accidentally discard many correct ones. Fig. 7 middle shows the warping result by augmenting CPW with our weighted feature matches. We can see that our seam-guided weighting scheme has effectively removed most of the local distortions while facilitating the alignment around the estimated seam. Fig. 7 right shows the result of our warping method. It further preserves extracted curve and line structures across the entire image region.

4.4   Comparison with APAP [25]

We use the source code provided by the authors to obtain image alignment by APAP method [25], after which we apply our seam estimation for fair comparison. All the results are generated using default parameters. Some of the comparison results are given in Fig. 8. APAP tries to align as many feature matches as possible in the entire overlapping region without explicitly preserving salient scene structures. It suffers from local distortions in both overlapping

Fig. 8. Comparison with APAP [25]. Top: APAP's results. Bottom: SEAGULL's results.

and non-overlapping regions (green rectangle regions in Fig. 8) caused by feature matches with large parallax. Furthermore, as the APAP warp is decoupled from the seam estimation process, such local distortion can have negative impact on seam estimation. The estimated seam may accidentally pass through these distorted regions and generate broken structures (green circle regions in Fig. 8). Our adaptive feature weighting explicitly avoids using feature matches with large parallax or far away from the estimated seam of interest to minimize the undesired local distortions. Together with our novel structure-preserving warp, our final stitching results are visually much more appealing for the given examples.

4.5 Comparison with Zhang and Liu's Method [26]

Zhang and Liu's method [26] is currently the state-of-the-art for parallax-tolerant image stitching. Since the source code is not available, we only test our method on the datasets released by the authors. In most cases, SEAGULL generates visually comparable stitching results, and produces noticeably better ones on some examples. The complete comparison can be found in the supplementary material. Here we show examples with noticeable improvements to demonstrate the advantages of SEAGULL. In Zhang and Liu's method, the best homography is selected from various rough alignment candidates. Therefore, the final stitching seam without any further optimization may still be contaminated by large misalignment (Fig. 9 row 1), even though the stitching quality as a whole might seem acceptable. Furthermore, since salient scene structures like curves or lines are not explicitly preserved by their method, they are found distorted in some stitching results (Fig. 9 row 2-4). In comparison, our structure-preserving warp does not produce such artifacts on these examples.

4.6 Discussion

All our experiments are performed on a desktop computer with an Intel i7 CPU and 32 GB memory. For each alignment hypothesis, the seam-guided local alignment process takes about $3 \sim 4$ seconds. The proposed algorithm usually takes less than one minute to find the best stitching result without code optimization.
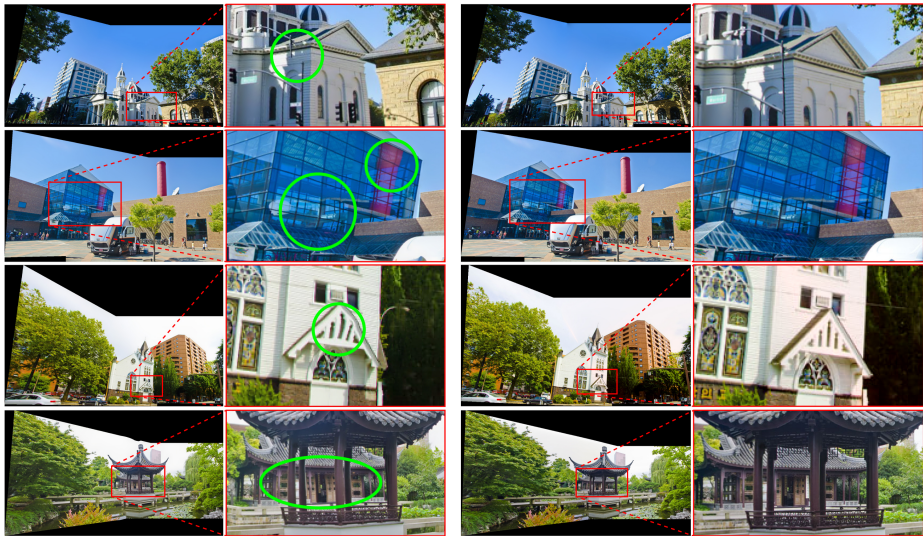
Fig. 9. Comparison with Zhang and Liu's method [26]. Left: Zhang and Liu's results. Right: SEAGULL's results.

Since the optimization for each alignment hypothesis is independent from one another, our method can be readily parallelized for better runtime. Our method could fail if the parallax is too large in the periphery of the overlapping region, or these local regions consist of rich salient structures but few feature matches.

5   Conclusions

In this paper, we propose a seam-guided local alignment method for large parallax image stitching. We closely couple the local alignment computation and the seam estimation via adaptive feature weighting. Salient curve and line structures are explicitly preserved during the warping by enforcing both local and non-local similarity constraints. Our superpixel-based feature grouping method effectively reduces the number of alignment hypotheses while still discovering good initial alignments for later optimization. The proposed method is evaluated on a variety of image pairs with large parallax and outperforms state-of-the-art stitching methods in terms of effectiveness and robustness.

6   Acknowledgements

References

1. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. 34(11), 2274–2282 (Nov 2012)
2. Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M.: Interactive digital photomontage. ACM Trans. Graph. 23(3), 294–302 (2004)
3. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. Int. J. Comput. Vision (IJCV) 74(1), 59–73 (2007)
4. Burt, P.J., Adelson, E.H.: A multiresolution spline with application to image mosaics. ACM Trans. Graph. 2(4), 217–236 (1983)
5. Carroll, R., Agarwala, A., Agrawala, M.: Image warps for artistic perspective manipulation. ACM Trans. Graph. 29(4), 127:1–127:9 (2010)
6. Chang, C.H., Sato, Y., Chuang, Y.Y.: Shape-preserving half-projective warps for image stitching. In: Proc. CVPR (2014)
7. Chang, C.H., Chuang, Y.Y.: A line-structure-preserving approach to image resizing. In: Proc. CVPR. pp. 1075–1082 (2012)
8. Farbman, Z., Hoffer, G., Lipman, Y., Cohen-Or, D., Lischinski, D.: Coordinates for instant image cloning. ACM Trans. Graph. 28(3), 67:1–67:9 (2009)
9. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6), 381–395 (1981)
10. Gao, J., Kim, S.J., Brown, M.S.: Constructing image panoramas using dual-homography warping. In: Proc. CVPR (2011)
11. Gao, J., Li, Y., Chin, T.J., Brown, M.S.: Seam-driven image stitching. In: Eurographics. pp. 45–48 (2013)
12. von Gioi, R.G., Jakubowicz, J., Morel, J.M., Randall, G.: Lsd: A fast line segment detector with a false detection control. IEEE Trans. Pattern Anal. Mach. Intell. 32(4), 722–732 (2010)
13. He, K., Chang, H., Sun, J.: Rectangling panoramic images via warping. ACM Trans. Graph. 32(4), 79:1–79:10 (2013)
14. Krähenbühl, P., Lang, M., Hornung, A., Gross, M.: A system for retargeting of streaming video. ACM Trans. Graph. 28(5), 126:1–126:10 (2009)
15. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: Image and video synthesis using graph cuts. ACM Trans. Graph. 22(3), 277–286 (2003)
16. Li, S., Yuan, L., Sun, J., Quan, L.: Dual-feature warping-based motion model estimation. In: Proc. ICCV. pp. 4283–4291 (2015)
17. Lin, C.C., Pankanti, S.U., Ramamurthy, K.N., Aravkin, A.Y.: Adaptive as-natural-as-possible image stitching. In: Proc. CVPR (2015)
18. Lin, K., Liu, S., Cheong, L.F., Zeng, B.: Seamless video stitching with hand-held camera inputs. Computer Graphics Forum 35(2), 479–487 (May 2016)
19. Lin, W.Y., Liu, S., Matsushita, Y., Ng, T.T., Cheong, L.F.: Smoothly varying affine stitching. In: Proc. CVPR (2011)
20. Liu, F., Gleicher, M., Jin, H., Agarwala, A.: Content-preserving warps for 3d video stabilization. ACM Trans. Graph. (Proc. of SIGGRAPH) 28(3), 44:1–44:9 (2009)
21. Liu, S., Yuan, L., Tan, P., Sun, J.: Bundled camera paths for video stabilization. ACM Transactions on Graphics (TOG) 32(4), 78:1–78:10 (2013)
22. Szeliski, R.: Image alignment and stitching: A tutorial. Found. Trends. Comput. Graph. Vis. 2(1), 1–104 (2006)

23. Szeliski, R., Shum, H.Y.: Creating full view panoramic image mosaics and environment maps. ACM Trans. Graph. (Proc. of SIGGRAPH) pp. 251–258 (1997)
24. Wu, C.: SiftGPU: A GPU implementation of scale invariant feature transform (SIFT). http://cs.unc.edu/ ccwu/siftgpu (2007)
25. Zaragoza, J., Chin, T.J., Brown, M.S., Suter, D.: As-projective-as-possible image stitching with moving DLT. In: Proc. CVPR (2013)
26. Zhang, F., Liu, F.: Parallax-tolerant image stitching. In: Proc. CVPR (2014)