# Video Stitching with Spatial-Temporal Content-Preserving Warping

Wei Jiang
Huawei Media Lab
Futurewei Technologies Inc.
wei.jiang@huawei.com

Jinwei Gu
Huawei Media Lab
Futurewei Technologies Inc.
jinwei.gu@huawei.com

## Abstract

*We propose a novel algorithm for stitching multiple synchronized video streams into a single panoramic video with spatial-temporal content-preserving warping. Compared to image stitching, video stitching faces several new challenges including temporal coherence, dominate foreground objects moving across views, and camera jittering. To overcome these issues, the proposed algorithm draws upon ideas from recent local warping methods in image stitching and video stabilization. For video frame alignment, we propose spatial-temporal local warping, which locally aligns frames from different videos while maintaining the temporal consistency. For aligned video frame composition, we find stitching seams with 3D graphcut on overlapped spatial-temporal volumes, where the 3D graph is weighted with object and motion saliency to reduce stitching artifacts. Experimental results show the advantages of the proposed algorithm over several state-of-the-art alternatives, especially in challenging conditions.*

## 1. Introduction

Stitching multiple synchronized video streams into a single panoramic video becomes increasingly important nowadays, given the wide applications of high definition, 360-degree videos such as wide area video surveillance, teleconferencing and tele-presence, and immersive virtual reality and augmented reality experiences.

One possible misconception many people may have is that video stitching is a somewhat *solved* problem. This is because of the success of image stitching algorithms [16] in generating panoramas from images taken from the same viewpoint or about a roughly planar scene. However, video stitching is, actually, a much more challenging problem [17], especially for non-ideal inputs (*e.g.*, the optical centers of the cameras are not exactly at the same location, the scene is non-planar, and/or dominate foreground objects move across cameras). Figure 1 gives an example of stitching such non-ideal inputs where problems can be clearly seen. In the example, three video cameras[1] are stitched together using AutoStitch (from OpenCV, which implements the work of [16]) with each frame stitched separately or using a single common alignment for all frames, a latest commercial software VideoStitchStudio2 [1], the re-

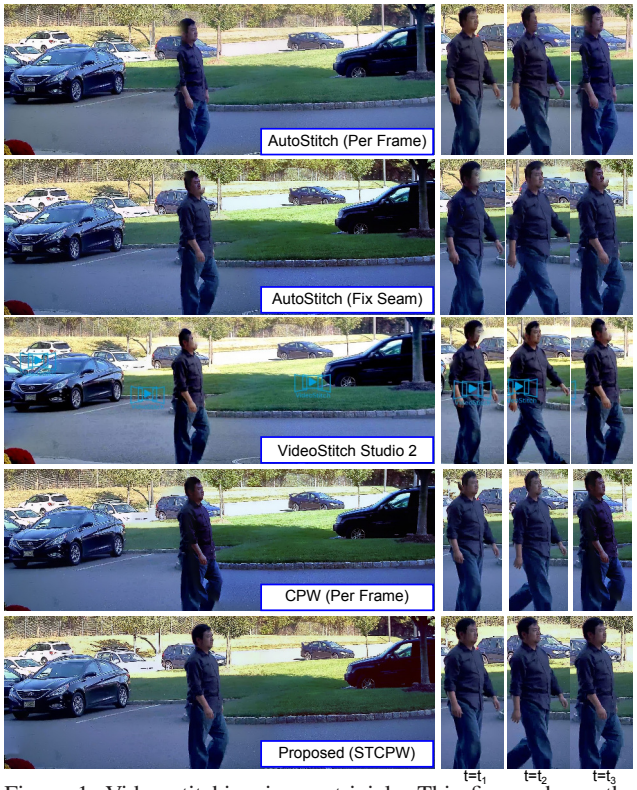


Figure 1. Video stitching is non-trivial. This figure shows the stitching of three video cameras, with AutoStitch from OpenCV (each frame stitched separately or using a single common alignment for all frames), a commercial software VideoStitchStudio2 [1], parallax-tolerant stitching (*i.e.*, CPW) for each frame separately [20], and our proposed method (*i.e.*, spatial-temporal content preserving warping STCPW). **Left**: one of the stitched frames. **Right**: zoomed insets over multiple frames. Please refer to the supplementary material for stitched videos.

[1]Full resolution images are embedded in this PDF. Please zoom in for viewing details, or refer to the supplementary materials.

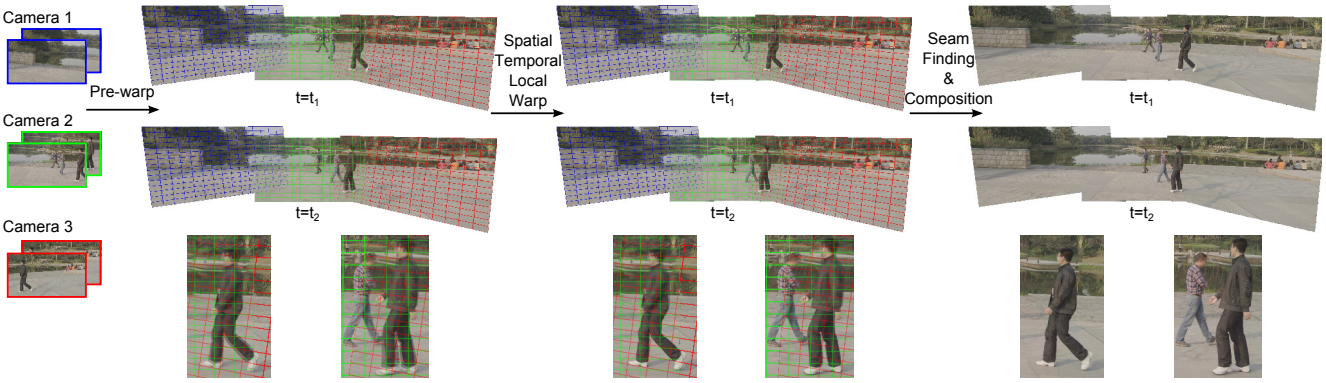


Figure 2. An overview of the proposed video stitching algorithm. Video frame alignment is done in two steps: a pre-warp with spatial and temporal global transformations, and a spatial-temporal local warping with STCPW. After alignment, a weighted 3D graphcut is used to find optimal seams (*i.e.*, 2D surfaces) in the overlapped volumes of the aligned videos for composition. The zoomed insets show that STCPW effectively reduces mis-alignment (*i.e.*, ghosting) after the pre-warp for dominate foreground moving objects.

cent parallax-tolerant image stitching method (*i.e.*, content-preserving warping or CPW) working on each frame separately [20], and our proposed method. As shown, video stitching is non-trivial — not only do we have to handle parallax as in challenging image stitching scenarios [19, 20], we also have to make stitching consistent over time. Moreover, camera jittering often adds additional complexity for video stitching, *e.g.*, for outdoor surveillance cameras, hand-held camcorders, or mobile phone cameras.

Recent advances in image stitching shows that *local stitching* is effective to deal with parallax problem in image alignment [20] for challenging scenarios with non-concentric cameras and non-planar scenes. In such methods, a global homography transformation handles global alignment and local CPW [11] adjusts the alignment in local regions to handle parallax issues. Interestingly, a similar *local* approach [12] based on CPW has also been successfully used in video stabilization by optimizing a bundle of camera paths (*i.e.*, one camera path per image grid).

Built upon the success of prior work, in this paper, we propose a novel video stitching algorithm, which performs *local warping and composition* in both spatial and temporal domains based on content-preserving warping. For video frame alignment (Sec. 3.1), we propose a spatial-temporal content-preserving warping (STCPW) algorithm that simultaneously optimizes a homography mesh per frame per camera by minimizing spatial and temporal matching costs. STCPW locally adjusts the alignment of frames from different videos while maintaining the temporal consistency of the alignment. For aligned video frame composition (Sec. 3.2), we formulate it as a weighted 3D graphcut problem within the overlapped spatial-temporal volumes of the aligned videos. Pixels around salient and moving objects are set with higher weights to avoid being cut through, which further reduces stitching artifacts.

We evaluate the proposed algorithm over two different multi-camera rigs, a high-end RED camera rig and a consumer-grade PointGrey Cricket IP camera rig, with an emphasis of the challenging case where dominant foreground objects move across cameras. A number of videos were captured for testing, covering various types of foreground object motion. Experimental results in Section 4 shows that the proposed method consistently outperforms several state-of-the-art alternatives in stitching videos of challenging conditions.

## 2. Related Work

**Image Stitching** Image stitching is a well-studied, yet still active research area [3, 16]. Recent research focuses on spatially-varying warping algorithms [9, 19] and local stitching methods [20]. Our work extends the local stitching algorithms to the temporal domain for video stitching. There are also prior work on video mosaic [14], where the goal is to create a panoramic still image from a video. Our goal is to create a panorama video from multiple videos.

**Video Stitching** Compared to image stitching, there are only very limited prior work on video stitching. Most prior video stitching methods either use a fixed alignment from still images [1, 21], or conducts stitching frame by frame separately [5]. Shimizu *et al*. proposed to stitch videos with pure translation motion for sport events [15]. El-Saban *et al*. studied video stitching of free-moving mobile devices [6]. Xu and Mulligan [18] used multi-grid SIFT for acceleration. To the best of our knowledge, ours is the first video stitching method that jointly optimizes the frame alignment and frame composition in both spatial and temporal domains to deal with non-ideal challenging videos.

**Video Stabilization** An area closely related to our work is video stabilization, To remove camera shakiness, in video stabilization, a smooth 2D or 3D camera motion is estimated to synthesize a stabilized video from an input video [7, 11]. Recent advances [12, 13] also show that a spatially-varying 2D camera motion (*e.g.*, one homography

per image region) is more effective to deal with parallax and other non-linear motion. Our work extends this idea and simultaneously optimizes multiple camera paths over time.

## 3. Proposed Algorithm

The proposed algorithm includes two parts, as shown in Fig. 2. The first part is video frame alignment, which is done in two steps: (1) pre-warp all frames with spatial and temporal global transformations for coarse geometric alignment; and (2) locally refine the geometric alignment by optimizing spatial-temporal local warps to minimize matching costs across cameras and over time. The second part is to composite the aligned video frames into a single panoramic video, where we formulate the spatial-temporal seam finding as a weighted 3D graphcut problem on overlapped spatial-temporal volumes, in which salient regions (*e.g.*, faces, foreground objects with dominate motion) are set with higher weights to avoid being cut through.

### 3.1. Frame Alignment of Multiple Videos

Given a set of $N$ input videos (each with $T$ frames), $I_{i,t}$, $i = 1, \cdots, N$, $t = 1, \cdots, T$, the goal is to find a warping map for each video frame so that all the frames of all videos will be aligned to a common reference canvas. A set of feature points $\{P_{i,t,k}\}$ are extracted from each frame $I_{i,t}$, and correspondences between these feature points are established. We used SIFT features in our work. Please refer to [16] for a survey on feature choices for image stitching.

#### 3.1.1 Pre-warping with Global Transformation

Based on the matched feature points, we first compute a spatial global homography transformation $H_i^S$ for each camera using the first $K$ frames $I_{i,t}$, $i = 1, \cdots, N$, $t = 1, \cdots, K$. This initial global alignment defines the common reference frame for the remaining video frames. We use all matched features in the first $K$ frames and RANSAC for computing $H_i^S$, in order to alleviate the instability of the computed $H_i^S$ caused by both camera movements and object movements in the scene. Alternatively, one can use a selective subset of features [20] for the initial global alignment when large scene parallax is present.

To make the global alignment smooth over time, we also need to compute a temporal global homography transformation $H_{i,t}^T$ for each frame to align with the corresponding reference frame, $i = 1, \cdots, N$, $t = 2, \cdots, T$. This step is similar to video stabilization, in which we aim to find a smooth path for each camera to the reference frame so that the final stitched video will be stable. This is beneficial especially when the multi-camera rig is moving or jittering during the capture. To compute $H_{i,t}^T$, we first compute the average of the homography transformations between consecutive frames $\bar{A}_i^T = \frac{1}{T-1} \sum_{t=2}^{T} A_i^T(t-1, t)$ to account for the global camera movement, where $A_i^T(t-1, t)$ is the homography between $I_{i,t-1}$ and $I_{i,t}$. $\bar{A}_i^T$ can be viewed as the target camera motion for camera $i$. Here we assumed a constant target camera motion and thus $\bar{A}_i^T$ is an averaged homography. Other options for target camera motion path over time can also be used [7]. Given the target camera motion paths, we have $H_{i,t}^T = \bar{A}_i^T H_{i,t-1}^T \left(A_i^T(t-1, t)\right)^{-1}$, $t = 2, \cdots, T$, and $H_{i,1}^T$ equals to the identity matrix $I$.

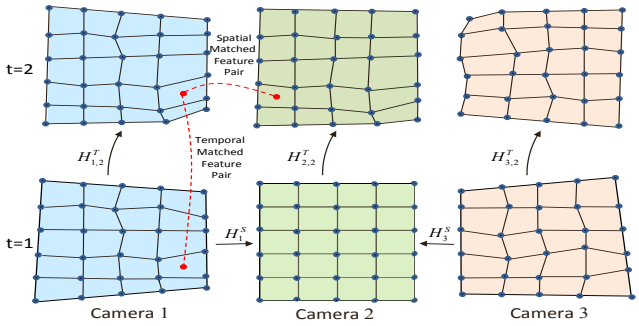

Figure 3. Spatial-temporal local warping is done by simultaneously optimizing vertices $\{\hat{V}_{i,t,k}\}$ of homography meshes for all frames of all cameras. First, pre-warping is done for each $I_{i,t}$ and feature $P_{i,t,k}$ based on Eq. (1), with global spatial and temporal transformations $H_i^S$ and $H_{i,t}^T$, respectively. Second, the pre-warped $\{\bar{V}_{i,t,k}\}$ are optimized to obtain $\{\hat{V}_{i,t,k}\}$ using matched feature pairs across multiple cameras (*i.e.*, spatial matched feature pairs) and over time (*i.e.*, temporal matched feature pairs). In this figure, the first frame of Camera 2 is the reference frame.

As shown in Fig. 3, with the spatial global transformations $H_i^S$ and the temporal global transformations $H_{i,t}^T$, we pre-warp each image frame $I_{i,t}$ and its associated feature points $\{P_{i,t,k}\}$ to the reference frame as follows:

$$\bar{I}_{i,t} = H_i^S \cdot H_{i,t}^T \cdot I_{i,t}, \qquad \bar{P}_{i,t,k} = H_i^S \cdot H_{i,t}^T \cdot P_{i,t,k}. \quad (1)$$

#### 3.1.2 Spatial-Temporal Local Warping

After the global transformation, a spatial-temporal local warping is used to handle parallax in the video frames. As shown in Fig. 3, the warping map is represented as a mesh of homography transformations for each frame. Specifically, we uniformly divide each image frame into an $M_1 \times M_2$ grid. Let $V_{i,t,k}$ and $\bar{V}_{i,t,k}$, $k = 1, \cdots, (M_1+1)(M_2+1)$ denote the vertices of the grid mesh on image $I_{i,t}$ and the pre-warped image $\bar{I}_{i,t}$, respectively. Our goal is to simultaneously optimize the target vertices $\hat{V}_{i,t,k}$ on the reference canvas of all the $NT$ meshes so that the corresponding matched features $\{P_{i,t,k}\}$ in these image frames are well aligned and the shape of the meshes are best preserved. The objective function is defined as a linear combination of matching cost terms in the spatial domain and temporal domain as follows:

$$E = E_{ds} + w_1 E_{gs} + w_2 E_{ss} + w_3 E_{dt} + w_4 E_{gt} + w_5 E_{st}, \quad (2)$$

where $E_{ds}$, $E_{gs}$, $E_{ss}$ are the terms in the spatial domain similar to those in [20] that measure, respectively, the local alignment, the global alignment, and the mesh smoothness;

$E_{dt}$, $E_{gt}$, $E_{st}$ are the corresponding terms in the temporal domain; and $\{w_i\}_{i=1,\cdots,5}$ are the weights.

**Spatial local alignment term** $E_{ds}$

$$E_{ds}=\sum_{t=1}^{T}\sum_{i\neq j}^{N}\sum_{k\in S_{i,j,t}}||\sum_{c=1}^{4}\lambda_{i,t,k}(c)\hat{V}_{i,t,k}(c)-\tilde{P}_{j,t,k}||^2, \quad (3)$$

where $S_{i,j,t} = \{k|\left(\bar{P}_{i,t,k}, \bar{P}_{j,t,k}\right)\}$ is the set of matched features between camera $i$ and camera $j$ at frame $t$ (*i.e.*, spatial matched feature pairs), $\lambda_{i,t,k}(c)$ are the barycentric weights for representing the corresponding feature $\bar{P}_{i,t,k}$ with the four vertices $\bar{V}_{i,t,k}(c)$ of a quad that contains $\bar{P}_{i,t,k}$ in the mesh, and $\tilde{P}_{j,t,k}$ is $\bar{P}_{j,t,k}$ on the final reference frame. $\tilde{P}_{j,t,k}$ can be assumed known, if we solve the stitching problem sequentially, *i.e.*, by setting one frame as the reference frame, and stitch with the remaining frames one at a time.

**Spatial global alignment term** $E_{gs}$

$$E_{gs} = \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{k=1}^{K}\tau_{i,t,k}||\hat{V}_{i,t,k} - \bar{V}_{i,t,k}||^2, \quad (4)$$

where $\tau_{i,t,k} = 1$ if no matched feature is close to $\bar{V}_{i,t,k}$ (*e.g.*, within $r = 10$ pixels) and $\tau_{i,t,k} = 0$ otherwise. Therefore, $E_{gs}$ encourages the target vertex $\hat{V}_{i,t,k}$ to remain the same with the pre-warped $\bar{V}_{i,t,k}$ if there are no matched features in the local neighborhood to guide its refinement.

**Spatial mesh smoothness term** $E_{ss}$

$$E_{ss} = \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{k\in\Delta} w_s \cdot g(\hat{V}_{i,t,k}, \bar{V}_{i,t,k})^2, \quad (5)$$

where $\Delta$ is the set of vertex triplets of the mesh, $w_s$ is the spatial edge saliency of the triplet (set similar as in [11]), and function $g(\cdot)$ measures the triangle similarity [20]:

$$g(\hat{V}, \bar{V}) = ||\hat{V}_1 - (\hat{V}_2 + u(\hat{V}_3 - \hat{V}_2) + vR(\hat{V}_3 - \hat{V}_2)||, \quad (6)$$

where $\{\bar{V}_i\}_{i=1,2,3}$ and $\{\bar{V}_i\}_{i=1,2,3}$ are the three vertices of the vertex triplet in the pre-warped mesh and the final mesh, $R = \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$, and $u$ and $v$ are computed by solving

$$\bar{V}_1 = \bar{V}_2 + u(\bar{V}_3 - \bar{V}_2) + vR(\bar{V}_3 - \bar{V}_2). \quad (7)$$

Minimizing $E_{ss}$ encourages the mesh to undergo a similarity transformation, which reduces local distortion.

**Temporal local alignment term** $E_{dt}$

$$E_{dt}=\sum_{t=2}^{T}\sum_{i=1}^{N}\sum_{k\in S_{i,t-1,t}}||\sum_{c=1}^{4}\lambda_{i,t,k}(c)\hat{V}_{i,t,k}(c)-\tilde{P}_{i,t-1,k}||^2, \quad (8)$$

where $S_{i,t-1,t}=\{k|\left(\bar{P}_{i,t-1,k}, \bar{P}_{i,t,k}\right)\}$ is the set of matched features between frame $t-1$ and $t$ for camera $i$ (*i.e.*, temporal matched feature pairs), and $\tilde{P}_{i,t-1,k}$ is $\bar{P}_{i,t-1,k}$ on the final reference frame. Similar to Eq. (3), $\tilde{P}_{i,t-1,k}$ can be assumed known, if we solve the stitching problem sequentially. Minimizing $E_{dt}$ encourages original frames $I_{i,t}$ to align with the corresponding reference frame for camera $i$.

**Temporal global alignment term** $E_{gt}$

$$E_{gt}=\sum_{t=2}^{T}\sum_{i=1}^{N}\sum_{k=1}^{K}\sigma_{i,t,k}||\hat{V}_{i,t-1,k}-\hat{V}_{i,t,k}||^2, \quad (9)$$

where $\sigma_{i,t,k}$ is a non-negative weight that is linearly proportional to the scale of pixel movement in the neighborhood of $\bar{V}_{i,t,k}$. Intuitively, if the neighborhood region of the pre-warped vertex $\bar{V}_{i,t,k}$ remains static over time, the corresponding vertex $\hat{V}_{i,t,k}$ should remain unchanged through time, and thus $\sigma_{i,t,k}$ should be larger, and vice versa. We set $\sigma_{i,t,k}$ to be the average distance between matched features within the neighborhood of $\bar{V}_{i,t,k}$ and normalize to $[0, 1]$. Unlike Eq (4), both $\hat{V}_{i,t-1,k}$ and $\hat{V}_{i,t,k}$ are unknowns.

**Temporal mesh smoothness term** $E_{st}$

$$E_{st}=\sum_{t=2}^{T}\sum_{i=1}^{N}\sum_{k\in\Delta} w_t \cdot g(\hat{V}_{i,t,k}, \bar{V}_{i,t-1,k})^2, \quad (10)$$

where the function $g(\cdot)$ is the same as defined in $E_{ss}$ and $w_t$ is the temporal edge saliency for each triplet (defined similarly as $w_s$). Minimizing $E_{st}$ encourages the mesh of each camera undergoes similarity transformation over time to reduce distortion.

**Intuitions of The Six Terms** The six terms above constrain the video stitching problem similar way to that in image stitching in [20]. The local alignment terms, $E_{ds}$ and $E_{dt}$, are the *data* terms, which adjust local homography based on matched features to avoid parallax. The global alignment terms, $E_{gs}$ and $E_{gt}$, are designed for areas without many matched features — they enforce the local warping to be stable in those areas. $E_{ss}$ and $E_{st}$ are the *smoothness* terms to prevent extreme distortion. The relative importance of these terms depends on scene content, camera layout, and user preference. Their weights are given below. Please note all the six terms are in fact quadratic functions and can be solved efficiently with linear least square. Details are shown below.

**STGlobal:** $M_1 = M_2 = 1$ One important special case is when $M_1 = M_2 = 1$, *i.e.*, we optimize only one quad (*i.e.*, one global homography transformation) per frame per camera. We refer this special case as STGlobal. Note that STGlobal uses the pre-warping output as its initial value, and then simultaneously optimizes the $NT$ global homography transformations based on all the matched feature pairs. As we will show later, in some cases, STGlobal is a good trade-off between computational costs, robustness, and stitched video quality. In general, if the homography meshes have more quads, it is more flexible to handle parallex problem, but it needs more computation to optimize, and it may be more vulnerable when the scene does not have sufficient matched features.

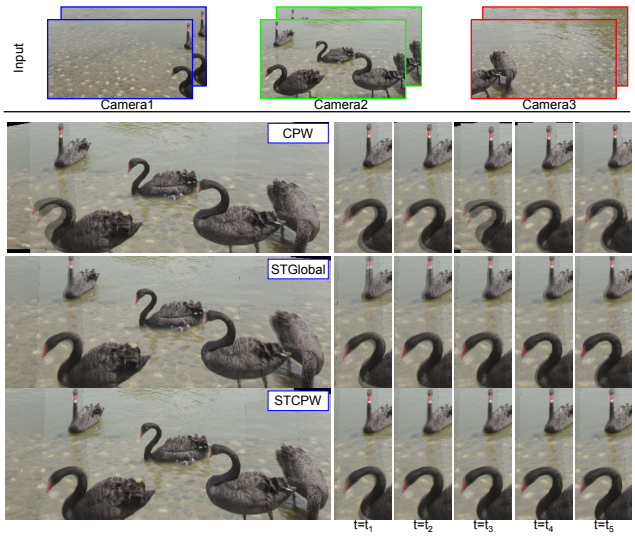


Figure 4. Frame alignment results of stitching three videos. No composition is done in order to visualize the quality of alignment only. **Top**: examples of input frames. **Bottom Left**: one of the stitched frames. **Bottom Right**: zoomed insets over multiple frames. As shown, CPW per frame can align part of the image well but exhibits large ghosting artifacts in other regions over time. STGlobal (*i.e.*, STCPW when $M_1 = M_2 = 1$) reduces temporal inconsistency, but both regions still have ghosting artifacts due to the limitation of global transformation. The proposed STCPW method has the least amount of ghosting consistently over time.

**Bundle Adjustment** To minimize the energy function in Eq. (2), we note that all the terms from Eq. (3) to Eq. (10) are quadratic functions of $\hat{V}_{i,j,k}$. Thus, if we solve it sequentially, *i.e.*, by setting one frame as the reference and stitch with one other frame at a time, each step can be solved efficiently with linear least square. This procedure, however, may accumulate significant errors for stitching a large number of cameras or video frames.

Alternatively, we use the sequential method to obtain an initial solution, and perform bundle adjustment to simultaneously optimize $\hat{V}_{i,t,k}$ for all frames and all cameras. Specifically, to obtain the initial solution, we first choose a reference camera by building a connectivity graph among cameras based on matched features and picking the one with the highest degree, and then find a camera with the most matched features with the reference camera to stitch. We repeat this step sequentially until all cameras are visited. For bundle adjustment, we note that because of Eq. (3) and Eq. (8), this is an iterative procedure, since the quads containing matched features may change during each iteration. In practice, the bundle adjustment converges with fewer than five iterations. We use the CERES-Solver [2] for bundle adjustment. With $\hat{V}_{i,t,k}$ solved, we warp $\bar{I}_{i,t}$ to the final reference frame $\tilde{I}_{i,t}$.

In our experiments, we set $w_3 = 1$, $w_4 = 0.3w_1$, and $w_5 = 0.3w_2$. We set $w_1$ and $w_2$ empirically in a similar way as in CPW [20], and $w_1 = 0.5$, $w_2 = 0.1$ in the paper. We use the first $K = 10$ frames to compute $H_i^S$, and $M_1 \times M_2 = 15 \times 15$ meshes for all the experiments.

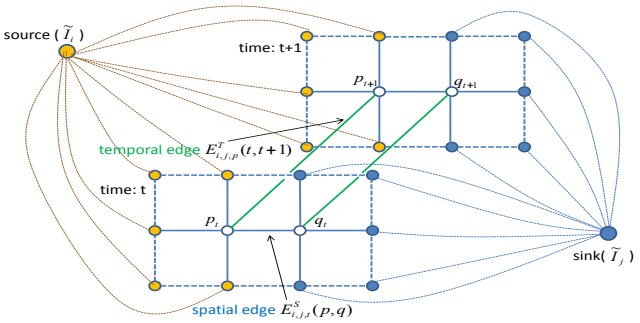


Figure 5. Spatial-temporal seam finding is formulated as a weighted 3D graph-cut problem over the overlapped space-time volume of two aligned videos. As shown, the four white nodes, $p_t$, $q_t$, $p_{t+1}$ and $q_{t+1}$, are of interests. They are linked with both spatial edges $E^S_{i,j,t}(p,q)$ and temporal edges $E^T_{i,j,p}(t,t+1)$, with each other and their neighboring pixels. The neighboring pixels are linked to one of the two videos with infinity cost. The goal is to find a minimal cut of the graph so that each white node can link to either the source $\tilde{I}_i$ or the sink $\tilde{I}_j$.

Figure 4 shows an example of frame alignment of stitching three videos. This example is challenging, due to the small overlapped regions and the small amount of salient features in foreground. As shown, CPW per frame can align part of the image well but exhibits large ghosting artifacts in other regions over time. STGlobal (*i.e.*, STCPW when $M_1 = M_2 = 1$) reduces temporal inconsistency, but still has ghosting artifacts due to the limitation of global transformation. The proposed STCPW method has the least amount of ghosting and it consistently outperforms others over time.

### 3.2. Spatial-Temporal Seam Finding

Next, we need to composite the aligned frames into a single panoramic video. We again consider all the frames together and perform spatial-temporal seam finding. As shown in Fig. 5, we formulate the spatial-temporal seam finding as a graph-cut problem over a space-time volume, similar to video texture synthesis [8]. Moreover, we assign higher weights to spatial edges and temporal edges that contains salient features (*e.g.*, faces, pedestrian) so that these regions will not be cut through.

Specifically, we construct a graph where each node is a pixel in the overlapped regions of all the aligned frames. There are two types of edges between each pair of graph nodes: spatial edges and temporal edges. The spatial edge is the edge between two graph nodes that corresponds to pixels at the same time index but different cameras. The temporal edges is the edge between two graph nodes that corresponds to pixels at the same camera but different time indices. The cost of a spatial edge between pixels $p$ and $q$ from camera $i$ and camera $j$ at time $t$ is defined as:

$$E^S_{i,j,t}(p,q) = o_p D(\tilde{I}_{i,t}(p), \tilde{I}_{j,t}(p)) + o_q D(\tilde{I}_{i,t}(q), \tilde{I}_{j,t}(q)), \quad (11)$$


Figure 6. Composition results for stitching three aligned videos. **Top**: we use the standard seam finding and blending method in AutoStitch in OpenCV, which is based on 2D graphcut. **Bottom**: we use the proposed spatial-temporal seam finding method, which is a 3D weighted graphcut method. The alignment is the same for both methods. Each row shows two stitched frames, with zoomed insets. As shown, the standard 2D seam method has artifacts and temporal inconsistency while the proposed 3D seam method works well.

where $D(\cdot)$ is the distance measurement between pixel value $\tilde{I}_{i,t}(p)$ and $\tilde{I}_{j,t}(p)$, and $o_p$ and $o_q$ are the weights for object saliency. The cost of temporal edges $E^T_{i,j,p}(t, t+1)$ is defined similarly, as the weighted combination of the distance between the pixel values of the corresponding pixels in two consecutive frames. In our experiments, $D(\cdot)$ is set simply as the sum of squared difference (SSD), and the object saliency $o$ is set by first performing face detection and motion detection and then setting high saliency for regions with face or dominate motion.

After the graph is constructed, the standard max-flow algorithm is performed to find the optimal labeling of the pixels in the overlapped volume of the two videos. When more than two videos are being stitched, this process is conducted by adding one video at a time to the stitched result. After seam finding, we used the same multi-band blending and color correction procedures from AutoStitch in OpenCV.

Figure 6 shows the composition results for stitching three aligned videos. We compared with the standard seam finding method in AutoStitch in OpenCV, which is 2D graphcut. The alignment is the same for both methods. As shown, the standard 2D seam finding method has artifacts and temporal inconsistency while the proposed 3D (*i.e.*, spatial-temporal) seam finding method works well.

Since both frame alignment and composition are done simultaneously in the spatial and temporal domains, it may require large amount of memory and computation time for processing long videos. In practice, we use a sliding window of $(-L, L)$ frames around each frame for such computation. Larger values of $L$ mean stronger spatial/temporal smoothing but require more computation and memory. Specifically, we used $L = 1 \sim 3$ in the paper. The computation cost of $L=1$ is only about $20\%$ more than individual frame processing, and it is often good enough to largely improve stitching quality.

## 4. Experimental Results

Since there is no publicly available video stitching benchmark data, we evaluated the algorithms on several sets of videos we captured. We have two multi-camera rigs with different FoV, image quality, and resolution. One rig consists of three PointGrey Cricket cameras, each capturing 1080p ($1920 \times 1080$) video at 30fps. The second rig consists of three RED Scarlet Dragon cameras, each capturing 4K ($4096 \times 2160$) video at 60fps. All these cameras are synchronized for video capture. Our test videos covers a set of challenging cases (*i.e.*, camera not co-centered, dominant foreground objects moving across cameras): horizontal single person motion, multiple person motion, and person moving towards cameras with scale change.


Figure 7. Video stitching results of three PointGrey Cricket cameras. **Left**: one of the stitched frames. **Right**: zoomed insets over multiple frames. Note the distortion and cut-through artifacts on the foreground moving object. Please refer to the supplementary material for stitched videos.

We compare the proposed algorithm with three methods: the baseline of running AutoStitch in OpenCV for each time instance separately (we also tried AuthStitch with

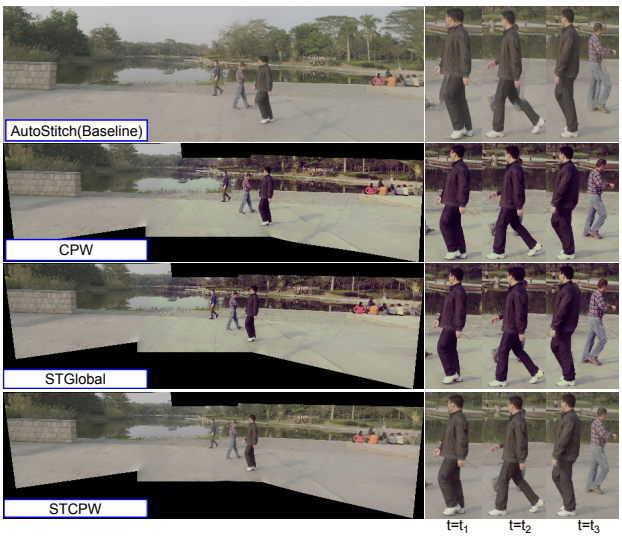


Figure 8. Video stitching results of three RED Scarlet Dragon 4K cameras. **Left**: one of the stitched frames. **Right**: zoomed insets over multiple frames. Note the ghosting and distortion artifacts on the foreground moving objects. Please refer to the supplementary material for stitched videos.

fixed seam but it often has more artifacts than the baseline); CPW per frame (performing local warping [20] for each time instance separately); and STGlobal (*i.e.*, STCPW when $M_1 = M_2 = 1$). CPW is one of the best methods for still image stitching, and comparing with it shows the benefit of joint spatial-temporal alignment. Comparing with STGlobal shows the benefit of local warping. All methods have the same color correction and blending steps from AutoStitch in OpenCV after seam finding.

Figures 7 and 8 show two examples of video stitching results. Please refer to the supplementary material for stitched videos and more results. As shown, the baseline method has severe ghosting artifacts. CPW per frame effectively reduces ghosting, but the stitched video is not stable since there is no temporal constraint enforced. STGlobal is more consistent over time, but still has ghosting in some frames due to the limitation of global transformation. The proposed STCPW consistently outperforms these methods.

## 5. Limitations and Discussions

We proposed a novel video stitching algorithm, which draws upon ideas from recent advances in parallax tolerant image stitching and video stabilization to perform spatial-temporal local warping and seam finding. Experimental results show its effectiveness for handing parallax and dominate foreground object moving problems.

The proposed algorithm is a first step for seamless video stitching. It has several limitations. (1) Like most image stitching algorithms, it relies on matched feature points for local warping. When foreground moving objects have few matched feature points, explicit foreground object detection/tracking maybe helpful. (2) The relative positions of the cameras are fixed and the algorithm can handle certain level of jittering. Video stitching of multiple freely moving cameras would require a dynamic definition of the reference frame. (3) The algorithm does not address the issue of stitching videos with large appearance differences, *e.g.*, exposures, color, depth of fields, *etc*. Existing solutions for still images [4, 10] can not be easily applied to videos because of the temporal consistency challenge. This will be another direction of our future work.

## References


[1] VideoStitchStudio V2 http://www.video-stitch.com/. 1, 2

[2] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 5

[3] M. Brown and D. Lowe. Automatic panoramic image sitching using invariant features. In *Proc. IEEE Int'l Conf. on Computer Vision (ICCV)*, 2007. 2

[4] A. Eden, M. Uyttendaele, and R. Szeliski. Seamseam image stitching of scenes with large motions and exposure differences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2006. 7

[5] M. El-Saban and et al. Stitching videos streamed by mobile phones in real-time. In *ACM Multimedia*, 2009. 2

[6] M. El-Saban and et al. Improved optimal seam selection blending for fast video stitching of videos captured from freely moving devices. In *Proc. IEEE Int'l Conf. on Image Processing (ICIP)*, 2011. 2

[7] M. Grundmann and et al. Autodirected video stabilization with robust L1 optimal camera paths. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2, 3

[8] V. Kwatra and et al. Graphcut textures: Image and video synthesis using graph cuts. *ACM Trans. on Graphics (SIGGRAPH)*, 2003. 5

[9] W.-Y. Lin and et al. Smoothly varying affine stitching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[10] W.-Y. Lin and et al. Aligning iimage in the wild. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012. 7

[11] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3d video stabilization. *ACM Trans. on Graphics (SIGGRAPH)*, 28, 2009. 2, 4

[12] S. Liu, L. Yuan, P. Tan, and J. Sun. Bundled camera paths for video stabilization. *ACM Trans. on Graphics (SIGGRAPH)*, 32(4):78, 2013. 2

[13] S. Liu, L. Yuan, P. Tan, and J. Sun. Steadyflow: Spatially smooth optical flow for video stabilization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[14] R. Marzotto, A. Fusiello, and V. Murino. High resolution video mosaicing with global alignment. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2004. 2

[15] T. Shimizu and et al. A fast video sitching method for motion-compensated frames in compressed video streams. In *International Conference on Consumer Electronics*, 2006. 2

[16] R. Szeliski. *Handbook of Mathematical Models in Computer Vision*, chapter Image Alignment and Sitching, pages 273–292. Springer, 2004. 1, 2, 3

[17] W. Xu. *Panoramic Video Stitching*. PhD thesis, University of Colorado at Boulder Boulder, 2012. 1

[18] W. Xu and J. Mulligan. Panoramic video stitching from commodity HDTV cameras. *Multimedia Systems*, 19(5):407–426, 2013. 2

[19] J. Zaragoza and et al. As-projective-as-possible image stitching with moving DLT. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 2013. 2

[20] F. Zhang and F. Liu. Parallax-tolerant image stitching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1, 2, 3, 4, 5, 7

[21] M. Zheng and et al. Stitching video from webcams. In *Proc. the 4th Int'l Symposium on Visual Computing (ISVC)*, 2008. 2