Correspondence Insertion for As-Projective-As-Possible Image Stitching

William X. Liu, Tat-Jun Chin
School of Computer Science
The University of Adelaide

Abstract

Spatially varying warps are increasingly popular for image alignment. In particular, as-projective-as-possible (APAP) warps have been proven effective for accurate panoramic stitching, especially in cases with significant depth parallax that defeat standard homographic warps. However, estimating spatially varying warps requires a sufficient number of feature matches. In image regions where feature detection or matching fail, the warp loses guidance and is unable to accurately model the true underlying warp, thus resulting in poor registration. In this paper, we propose a correspondence insertion method for APAP warps, with a focus on panoramic stitching. Our method automatically identifies misaligned regions, and inserts appropriate point correspondences to increase the flexibility of the warp and improve alignment. Unlike other warp varieties, the underlying projective regularisation of APAP warps reduces overfitting and geometric distortion, despite increases to the warp complexity. Comparisons with recent techniques for parallax-tolerant image stitching demonstrate the effectiveness and simplicity of our approach.

1. Introduction

The standard pipeline used in panoramic image stitching [20, 6] begins by detecting and matching local features or keypoints across the input images. A robust technique such as RANSAC is invoked to estimate the alignment functions, usually projective transformations (i.e. homographies), based on the feature matches. Bundle adjustment is then conducted to refine the homographies, before blending and compositing of the overlap pixels take place.

The usage of homographic warps for image stitching has been questioned [10, 23], since it carries the assumptions that the images were taken under pure rotational motions, or that the scene is sufficiently far away such that it is effectively planar - conditions unlikely to be satisfied in casual photography. As a result, misalignment effects or "ghosting" inevitably occur, and relatively costly postprocessing routines are necessary to rectify or conceal the errors.

(a) Input images with verified keypoint correspondences.

(b) Image stitching result using APAP warp.

(c) Result after automatically optimizing 25 new correspondences (indicated as yellow crosses) using our method.

Figure 1. (a) Two input images with verified keypoint correspondences; (b) Although APAP warp is spatially varying, without sufficient keypoint correspondences, the flexibility of the warp cannot be realised and the overlap area cannot be aligned well; (c) Our proposed correspondence insertion algorithm automatically inserts and optimizes new correspondences to improve the alignment.

Spatially varying warps have been proposed as alternatives to homographic warps [16, 10, 23, 9]. Such warps can better account for the effects of parallax when aligning the overlap regions. In particular, as-projective-as-possible (APAP) warps [23] interpolate the data flexibly, while maintaining a global projective trend so as to extrapolate correctly. Half-projective warps [9] improve upon APAP by preventing excessive stretching when extrapolating.

Ultimately, spatially varying warps are only as flexible as warranted by available feature matches. Without a sufficiently dense sampling of the underlying interpolant, the warp reduces to the baseline warp (similarity [16], projective [23]), thus defeating its spatially varying ability. A

large number of feature matches are thus required to obtain good alignment, especially in areas with parallax where the true alignment function deviates from a simple homography. There is no guarantee, however, that feature matches are produced uniformly in the overlap area; see Fig. 1.

A potential remedy is to use flow-based methods to obtain dense flow fields (e.g., [7, 14, 15]). However, these methods are geared towards feature tracking in videos for motion analysis and segmentation problems. Our tests show that flow-based methods often fail in producing accurate correspondences for wide-baseline image stitching, especially in scenes with repetitive textures; see Fig. 4.

On the other hand, direct methods [21, 19] align images based on pixel intensities, without requiring *a priori* established feature matches. Thus the problem of undersampling the true warp does not exist. Data-driven approaches to adaptively increase the warp complexity (e.g., by adding centers or control points in splines [8, 18, 5]) are also available. It is well known, however, that direct methods easily get stuck in local minima, thus necessitating coarse-to-fine registration strategies which can be inefficient [3].

An alternative idea is that perfect alignment throughout the overlap region is unnecessary [24]. Rather, images need only be aligned well in a local area, and a randomized algorithm was proposed to find a local homography. Seam cut [1] is then used to remove misalignments elsewhere. Such an approach is heavily dependent on *postprocessing* by seam cut. However, if misalignments are too severe, seam cut may not produce geometrically correct results [13]. This will occur when the true alignment function deviates significantly from a homography, e.g., when there are two apparent planes; see Fig. 2. More crucially, this method is reliant on the existing set of keypoint matches and cannot introduce new correspondences.

Contributions Our work differs from [24] in that we attempt to accurately align the images *throughout* the overlap area before compositing. Specifically, in correspondence-poor regions, we propose a *correspondence insertion* algorithm such that a good warping function can still be estimated. We show how correspondence search can be accomplished for moving direct linear transformation (MDLT), which is the estimation method for APAP warps [23]. We also highlight the simplicity of our data-driven warp adaptation scheme over previous spline-based center insertion techniques [8, 18, 5]. On panoramic mosaicing problems that are challenging, we show that our approach achieves accurate alignment without being handicapped by insufficient feature matches. Fig. 1 gives a preview of our method.

1.1. Previous work on center insertion

Center insertion has been studied extensively in spline regression [12, Chapter 5]. In particular, center insertion has been proposed for *pixel-based* non-rigid object registra-

(a) Result with simple linear blending (pixel averaging).

(b) Result after seam cut pixel selection to remove ghosting.
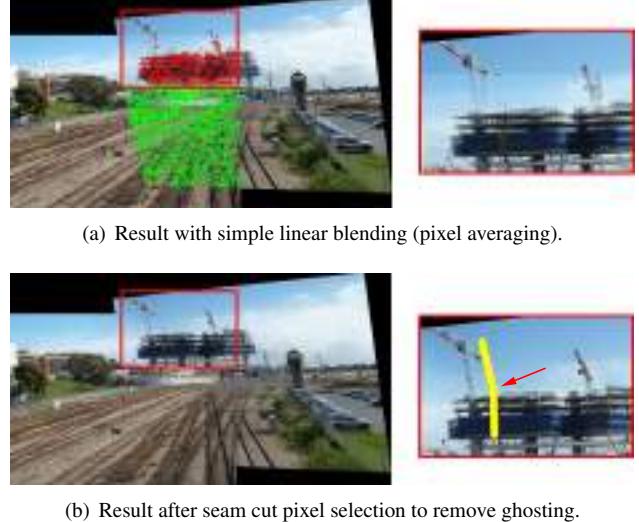
Figure 2. (a) Parallax-tolerant image stitching finds a homography that aligns a local region as well as possible. Here, green points are correspondences that are fitted by the local homography. Expectedly, regions that do not lie on the same plane cannot be aligned well; (b) Seam cut removes ghosting, but produces perceptually awkward results; note that the left crane appears to be bent. This result was taken directly from the project website of [24].

tion [8, 18, 5]. A 2D spline $f : \mathbb{R}^2 \mapsto \mathbb{R}^2$ is a function

$$f(\mathbf{x}) = \mathbf{A}^T \tilde{\mathbf{x}} + \sum_{k=1}^{K} \alpha_k \phi(\|\mathbf{x} - \mathbf{c}_k\|_2), \qquad (1)$$

where $\mathbf{A} \in \mathbb{R}^{3 \times 2}$ is an affine warp, $\tilde{\mathbf{x}} = [\mathbf{x}^T, 1]^T$ is $\mathbf{x}$ in augmented coordinates, $\{\alpha_k\}$ are scalar coefficients, $\{\mathbf{c}_k\}$ are 2D positions called centers, and ϕ is a radial basis function (RBF). The centers can be arbitrary (e.g., on a grid [21] over $\mathbb{R}^2$), and need not coincide with detected features.

The complexity of the warp increases with the number of centers K. If the pixels cannot be aligned well due to insufficient warp flexibility, one may consider adding new centers $\mathbf{c}_*$. Each insertion requires deciding where to place $\mathbf{c}_*$, and how to update the parameters $\{\mathbf{A}, \alpha_1, \dots, \alpha_K, \alpha_*\}$. W.r.t. the latter, in [5] the Gauss-Newton algorithm is used to adjust the parameters to further minimize the intensity difference in the overlap area. Note that the spline parameters are not independent, e.g., the coefficients in Thin Plate Splines (TPS) must satisfy the side condition $\sum_k \alpha_k = 0$. Thus, the updates get costlier as more centers are inserted.

Note that if $\mathbf{x}$ is sufficiently far away from all $\{\mathbf{c}_k\}$, the side condition and the monotonically decreasing RBF ensures that $f(\mathbf{x})$ reduces to the affinity $\mathbf{A}$. This implies that splines are unsuitable for image stitching, since ideally the warp should revert to a homography in the extrapolation areas [23]. While there exist splines with a projective baseline [4], the fact remains that parameter updating can be rel-

atively non-trivial. We show how the equivalent optimization on MLS regression is much simpler and more efficient.

2. Correspondence Search

Our goal is to find a warping function $f(\mathbf{x})$ that maps pixels from the source image I to the target image I'. A set of point-wise matches $\mathcal{X} = \{\mathbf{x}_i, \mathbf{x}'_i\}_{i=1}^N$ are first established across I and I', where $\mathbf{x}_i = [p_i, q_i]^T$ and $\mathbf{x}'_i = [p'_i, q'_i]^T$. The matches provide a sample of the true underlying warp, and we wish to estimate $f(\mathbf{x})$ from $\mathcal{X}$. In regions where $\mathcal{X}$ undersamples the true warp (e.g., insufficient point matches), the accuracy of $f(\mathbf{x})$ in approximating the true warp is limited. We wish to construct a method to generate new correspondences $\{\mathbf{x}_*, \mathbf{x}'_*\}$ to improve $f(\mathbf{x})$, given that $f(\mathbf{x})$ is modeled as an APAP warp [23].

This section describes a novel algorithm to optimize $\mathbf{x}'_*$ for a newly inserted $\mathbf{x}_*$. Sec. 3 presents a data-driven algorithm for choosing $\mathbf{x}_*$, in the context of panoramic stitching.

2.1. APAP warp

An APAP warp is defined by

$$f(\mathbf{x}) = \left[\frac{\mathbf{h}_1(\mathbf{x})^T \tilde{\mathbf{x}}}{\mathbf{h}_3(\mathbf{x})^T \tilde{\mathbf{x}}} , \frac{\mathbf{h}_2(\mathbf{x})^T \tilde{\mathbf{x}}}{\mathbf{h}_3(\mathbf{x})^T \tilde{\mathbf{x}}} \right]^T, \quad (2)$$

which is basically a projective warp (here, defined in inhomogeneous coordinates), but where the homography

$$\mathbf{H}(\mathbf{x}) = [\mathbf{h}_1(\mathbf{x}), \mathbf{h}_2(\mathbf{x}), \mathbf{h}_3(\mathbf{x})]^T \in \mathbb{R}^{3 \times 3} \quad (3)$$

is input-dependent, and $\mathbf{H}(\mathbf{x})$ is estimated using MDLT as

$$\mathbf{h}(\mathbf{x}) = \arg\min_{\mathbf{h}} \sum_{i=1}^N w_i(\mathbf{x}) \|\mathbf{m}_i \mathbf{h}\|_2^2, \quad \text{s.t. } \|\mathbf{h}\| = 1. \quad (4)$$

Here, $\mathbf{h}(\mathbf{x}) \in \mathbb{R}^9$ is the column-wise vectorized form of $\mathbf{H}(\mathbf{x})$, $\mathbf{m}_i \in \mathbb{R}^{2 \times 9}$ contains monomials from linearizing the homography constraint for the i-th datum $\{\mathbf{x}_i, \mathbf{x}'_i\}$

$$\mathbf{m}_i = \begin{bmatrix} \mathbf{0}_{1 \times 3} & -\tilde{\mathbf{x}}_i^T & q'_i \tilde{\mathbf{x}}_i^T \\ \tilde{\mathbf{x}}_i^T & \mathbf{0}_{1 \times 3} & -p'_i \tilde{\mathbf{x}}_i^T \end{bmatrix}, \quad (5)$$

and $w_i(\mathbf{x})$ is a *non-stationary* weight

$$w_i(\mathbf{x}) = \exp(-\|\mathbf{x} - \mathbf{x}_i\|_2^2 / 2\sigma^2). \quad (6)$$

Intuitively, (2) is a *moving average* of locally weighted projective warps, where σ in (6) controls the warp smoothness.

Equation (4) defines a weighted algebraic least squares problem, which can be rewritten in the matrix form

$$\mathbf{h}(\mathbf{x}) = \arg\min_{\mathbf{h}} \|\mathbf{W}(\mathbf{x})\mathbf{M}\mathbf{h}\|_2^2, \quad \text{s.t. } \|\mathbf{h}\| = 1, \quad (7)$$

where $\mathbf{W}(\mathbf{x})$ is a $2N \times 2N$ diagonal matrix containing

$$w_1(\mathbf{x}), w_1(\mathbf{x}), w_2(\mathbf{x}), w_2(\mathbf{x}), \ldots, w_N(\mathbf{x}), w_N(\mathbf{x}), \quad (8)$$

and $\mathbf{M}$ is a $2N \times 9$ matrix obtained by vertically stacking the monomial matrices $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_N$. The solution to (7) is the least significant eigenvector of $[\mathbf{W}(\mathbf{x})\mathbf{M}]^T \mathbf{W}(\mathbf{x})\mathbf{M}$.

2.2. Objective function and minimization

In regions with sparse correspondences, $\mathbf{x}$ is equally far (relative to σ) from all $\{\mathbf{x}_i\}_{i=1}^N$, and $f(\mathbf{x})$ reduces to a "rigid" projective warp, thus losing its spatially varying ability. Let $\mathbf{x}_*$ be a newly inserted point in I to raise the flexibility of $f(\mathbf{x})$ (again, selecting $\mathbf{x}_*$ will be discussed in Sec. 3). In the absence of geometric information, we need to rely on pixel intensity values to find a matching point $\mathbf{x}'_*$ in I'. To this end, we define the intensity matching cost

$$E(\mathbf{x}'_*) = \sum_{\mathbf{x} \in \mathbb{D}} \left[I'(f(\mathbf{x}|\mathbf{x}'_*)) - I(\mathbf{x}) \right]^2, \quad (9)$$

where $I(\mathbf{x})$ is the pixel intensity at $\mathbf{x}$, $\mathbb{D}$ is a region in I (by default, $\mathbb{D}$ is a 31×31 subwindow), and the warp $f(\mathbf{x}|\mathbf{x}'_*)$ is now dependent on $\mathbf{x}'_*$. Specifically, the input dependent homography is now obtained as

$$\mathbf{h}(\mathbf{x}|\mathbf{x}'_*) = \arg\min_{\mathbf{h}} \sum_{i=1}^N w_i(\mathbf{x}) \|\mathbf{m}_i \mathbf{h}\|_2^2 + w_*(\mathbf{x}) \|\mathbf{m}_*(\mathbf{x}'_*)\mathbf{h}\|_2^2,$$
$$\text{s.t. } \|\mathbf{h}\| = 1, \quad (10)$$

where $\mathbf{m}_*(\mathbf{x}'_*)$ contains the monomials for $\{\mathbf{x}_*, \mathbf{x}'_*\}$, and

$$w_*(\mathbf{x}) = \exp(-\|\mathbf{x} - \mathbf{x}_*\|_2^2 / 2\sigma^2). \quad (11)$$

In matrix form, (10) can be rewritten as

$$\mathbf{h}(\mathbf{x}|\mathbf{x}'_*) = \arg\min_{\mathbf{h}} \|\mathbf{W}_*(\mathbf{x})\mathbf{M}(\mathbf{x}'_*)\mathbf{h}\|_2^2,$$
$$\text{s.t. } \|\mathbf{h}\| = 1, \quad (12)$$

where $\mathbf{W}_*(\mathbf{x})$ is $\mathbf{W}(\mathbf{x})$ diagonally extended with two $w_*(\mathbf{x})$ values, and $\mathbf{M}(\mathbf{x}'_*)$ is $\mathbf{M}$ vertically appended with $\mathbf{m}_*(\mathbf{x}'_*)$. Note that only $\mathbf{M}(\mathbf{x}'_*)$ contains the variable $\mathbf{x}'_*$.

Our aim is to find $\mathbf{x}'_*$ by minimizing (9). We apply the well-known Lucas-Kanade (LK) technique [3]. A first-order Taylor expansion is applied on $E(\mathbf{x}'_* + \Delta\mathbf{x}'_*)$ to yield

$$\sum_{\mathbf{x} \in \mathbb{D}} \left[I'(f(\mathbf{x}|\mathbf{x}'_*)) + \nabla I'(f(\mathbf{x}|\mathbf{x}'_*)) \frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} \Delta\mathbf{x}'_* - I(\mathbf{x}) \right],$$

where image gradient $\nabla I'$ is computed using finite differencing. Differentiating against $\mathbf{x}'_*$ and equating to 0 yields

$$\Delta\mathbf{x}'_* = \mathbf{F}^{-1} \sum_{\mathbf{x} \in \mathbb{D}} \left[\nabla I'(f(\mathbf{x}|\mathbf{x}'_*)) \frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} \right]^T \left[I(\mathbf{x}) - I'(f(\mathbf{x}|\mathbf{x}'_*)) \right]$$
$$(13)$$

where $\mathbf{F}$ is the (approximated) Hessian

$$\mathbf{F} = \sum_{\mathbf{x} \in \mathbb{D}} \left[\nabla I'(f(\mathbf{x}|\mathbf{x}'_*)) \frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} \right]^T \left[\nabla I'(f(\mathbf{x}|\mathbf{x}'_*)) \frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} \right].$$

The current value for $\mathbf{x}'_*$ is then updated by $\Delta\mathbf{x}'_*$, and the steps are repeated until convergence. Refer to [3] for details.

To initialize $\mathbf{x}'_*$, we map $\mathbf{x}_*$ with the $f(\mathbf{x})$ prior to correspondence insertion. It is crucial to note that $\mathbf{h}(\mathbf{x}|\mathbf{x}'_*)$ changes for different $\mathbf{x} \in \mathbb{D}$. Essentially a unique homography is estimated for each $\mathbf{x} \in \mathbb{D}$ given $\mathbf{x}'_*$, thus realizing a spatially varying warp. This differs from the standard LK approach for "frame global" projective registration [3].

Brightness constancy assumption The objective function (9) assumes brightness constancy. In our context, this means corresponding pixels across input images have the same colour/brightness. This assumption may not hold in general, especially if the images are taken using cameras with various color auto-correction routines. To ensure the applicability of our method, we can apply color normalization techniques on the input images prior to stitching [22].

2.3. Jacobian of APAP warp

Evaluating $f(\mathbf{x}|\mathbf{x}'_*)$ and its Jacobian requires solving the weighted algebraic least squares problem (12) at each iteration - again, this differs from the common types of parametric motions used in LK [3]. Specifically, the solution $\mathbf{h}(\mathbf{x}|\mathbf{x}'_*)$ to (12) is the least significant eigenvector of

$$\mathbf{S}(\mathbf{x}|\mathbf{x}'_*) := [\mathbf{W}_*(\mathbf{x})\mathbf{M}(\mathbf{x}'_*)]^T[\mathbf{W}_*(\mathbf{x})\mathbf{M}(\mathbf{x}'_*)], \quad (14)$$

where $\mathbf{S}(\mathbf{x}|\mathbf{x}'_*)$ varies with $\mathbf{x}'_*$. The eigenvector satisfies

$$[\mathbf{S}(\mathbf{x}|\mathbf{x}'_*) - \lambda(\mathbf{x}|\mathbf{x}'_*)]\mathbf{h}(\mathbf{x}|\mathbf{x}'_*) = 0, \quad (15)$$
$$\|\mathbf{h}(\mathbf{x}|\mathbf{x}'_*)\| = 1, \quad (16)$$

where $\lambda(\mathbf{x}|\mathbf{x}'_*)$ is the eigenvalue. Via the chain rule,

$$\frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} = \frac{\partial f(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{h}(\mathbf{x}|\mathbf{x}'_*)} \frac{\partial \mathbf{h}(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*}. \quad (17)$$

The first term can be obtained by differentiating (2) - for brevity, we do not describe this simple process here.

The second term requires differentiating the eigenvector. Based on known results [17], the following expression

$$\frac{\partial \mathbf{h}(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*} = [\lambda(\mathbf{x}|\mathbf{x}'_*)\mathbf{I} - \mathbf{S}(\mathbf{x}|\mathbf{x}'_*)]^{\dagger} \frac{\partial \mathbf{S}(\mathbf{x}|\mathbf{x}'_*)}{\partial \mathbf{x}'_*}\mathbf{h}(\mathbf{x}|\mathbf{x}'_*) \quad (18)$$

can be derived, where $\mathbf{I}$ is the identity matrix. The derivative of $\mathbf{S}(\mathbf{x}|\mathbf{x}'_*)$ can in turn be obtained based on (14) - again, for brevity, we do not describe this simple process here. Note that only the last-two rows of $\mathbf{M}(\mathbf{x}'_*)$ depend on $\mathbf{x}'_*$.

Our correspondence search procedure is summarized in Algorithm 1. Note that in Step 4, the eigenvector $\mathbf{h}(\mathbf{x}|\mathbf{x}'_*)$ for each $\mathbf{x} \in \mathbb{D}$ needs to be calculated. Using modern linear algebra packages, this does not represent significant computational load, even for large $\mathbf{S}(\mathbf{x}|\mathbf{x}'_*)$, e.g., 1000×1000. Moreover, an incremental decomposition scheme [23] can be used to further reduce computational cost.

Algorithm 1 Correspondence search for APAP warp.

Require: Images I and I', feature matches $\{\mathbf{x}_i, \mathbf{x}'_i\}_{i=1}^N$, novel point $\mathbf{x}_*$.
1: Initialize $\mathbf{x}'_*$ by warping $\mathbf{x}_*$ using (2).
2: **repeat**
3:     **for** each $\mathbf{x} \in \mathbb{D}$ **do**
4:         Solve (12) to obtain $\mathbf{h}(\mathbf{x}|\mathbf{x}'_*)$ and $\lambda(\mathbf{x}|\mathbf{x}'_*)$.
5:         Calculate transformation $f(\mathbf{x}|\mathbf{x}'_*)$.
6:         Calculate warp Jacobian (17) for $\mathbf{x}$.
7:     **end for**
8:     Calculate $\Delta\mathbf{x}'_*$ (13) and update $\mathbf{x}'_* \leftarrow \mathbf{x}'_* + \Delta\mathbf{x}'_*$.
9: **until** $\mathbf{x}'_*$ converges.

Comparison against center insertion for splines At this juncture, it is instructive to compare our correspondence insertion algorithm with spline-based center insertion techniques (Sec. 1.1). Our warp update algorithm involves nothing more than searching for a 2D point $\mathbf{x}'_*$. This is a direct consequence of using "point set surfaces" [2] to define the warp. Contrast this to spline-based center insertion schemes, where all the warp parameters $\{\mathbf{A}, \alpha_1, \ldots, \alpha_K, \alpha_*\}$ need to be adjusted in each update.

3. Data-Driven Warp Adaptation

The previous section presented an algorithm that optimizes $\mathbf{x}'_*$ given $\mathbf{x}_*$. The remaining problem now is how to choose $\mathbf{x}_*$ to improve an APAP warp for image stitching. Our technique is encapsulated in a data-driven warp adaption scheme, which iteratively inserts new correspondences until sufficient "coverage" of the overlap area is achieved. Algorithm 2 summarises the method while Fig. 3 illustrates the core steps. Details are in the following.

Given the current correspondence set $\mathcal{X} = \{\mathbf{x}_i, \mathbf{x}'_i\}_{i=1}^N$, an APAP warp $f(\mathbf{x})$ (2) is first estimated and used to warp the source image I to align with the target image I'. Naturally we should strive to add correspondences in regions with high alignment errors. This is provided by the *absolute intensity difference* map R. Since we warp I to align with I', it is natural to put R in the same frame as I'. We ignore pixels (by zeroing the corresponding values in R) with error less than ϵ (default ϵ is 100). See Fig. 3(a).

Our approach relies on seam cut [1] for pixel selection during compositing; see Fig. 3(b). Therefore, since pixels that will have their color copied (more appropriately, retained) from I' are not subjected to misalignment errors, the corresponding values in R are zeroed. See Fig. 3(c).

Misalignments in regions with less structured textures (e.g., sky, trees, white board) are less obvious, thus it is less essential to introduce new correspondences in such locations. To realise this intuition, our scheme computes the visual saliency map of I' using the method of [11]; see Fig. 3(d). Values in R corresponding to pixels with saliency

Algorithm 2 Data-driven warp adaptation.

Require: Input images I and I', initial correspondence set $\mathcal{X} = \{\mathbf{x}_i, \mathbf{x}'_i\}_{i=1}^N$, error threshold ϵ, saliency threshold η, distance threshold ρ, and acceptance threshold ω.

1:   $\mathcal{L} \leftarrow \{\mathbf{x}'_i\}_{i=1}^N$.
2:   Compute visual saliency map on I'; see Fig. 3(d).
3:   **loop**
4:      Estimate APAP warp $f(\mathbf{x})$ from $\mathcal{X}$.
5:      Warp I to align with I' using $f(\mathbf{x})$.
6:      $R \leftarrow$ absolute intensity diff. map in overlap area.
7:      Set values in R which are $< \epsilon$ to 0; see Fig. 3(a).
8:      Optimize seam [1] for pixel selection in overlap area; see Fig. 3(b).
9:      Set values in R corresponding to pixels selected from I' according to the seam to 0; see Fig. 3(c).
10:     Set values in R corresponding to pixels of I' with saliency $< \eta$ to 0; see Fig. 3(e).
11:     $D \leftarrow$ distance transform on $\mathcal{L}$ in the overlap area.
12:     Set values in D that are $< \rho$ to ∞; see Fig. 3(f).
13:     If $D./R$ is all ∞, then break.
14:     $\mathbf{x}'_{\min} \leftarrow$ location in $D./R$ with minimum value.
15:     $\mathbf{x}_* \leftarrow f^{-1}(\mathbf{x}'_{\min})$.
16:     $\mathbf{x}'_* \leftarrow$ optimized correspondence from Algorithm 1.
17:     **if** $E(\mathbf{x}'_*) < \omega$ **then**
18:        $\mathcal{X} \leftarrow \mathcal{X} \cup \{\mathbf{x}_*, \mathbf{x}'_*\}$.
19:     **end if**
20:     $\mathcal{L} \leftarrow \mathcal{L} \cup \mathbf{x}'_{\min}$.
21: **end loop**

(a)   (b)   (c)   (d)   (e)   (f)

Figure 3. Data-driven warp adaptation. (a) Absolute difference map R with values $< \epsilon$ zeroed; (b) Optimized seam for the current $f(\mathbf{x})$; (c) Values in R corresponding to pixels selected from I' are zeroed; (d) Visual saliency map of I'; (e) Values in R corresponding to pixels with saliency $< \eta$ zeroed; (f) Distance transform D on $\{f(\mathbf{x}_i)\}_{i=1}^N$ with values $< \rho$ set to ∞ (darker areas here mean lower D values). Green cross indicates the $\mathbf{x}'_{\min}$ in this iteration.

less than η (default η is 0.5) are zeroed (recall that R has the same coordinate frame as I'); see Fig. 3(e).

At this stage, we have now produced an error map to guide the insertion of new correspondences. Additional constraints are given by the existing correspondence set $\mathcal{X}$. Specifically, we should insert new correspondences in regions that are not too near to $\mathcal{X}$, so as to avoid inserting redundant correspondences, and also not too far from $\mathcal{X}$, so as to ensure that correspondence search can be bootstrapped effectively by the existing $f(\mathbf{x})$. These constraints are realised by computing the distance transform D on the current set of features $\mathcal{L}$ in I'. Values of D that are less than ρ (default ρ is 15) are set to ∞; see Fig. 3(f).

Given D and R, the position $\mathbf{x}'_{\min}$ that has the lowest value in $D./R$ is sought, where "./" indicates element-wise division. The new point $\mathbf{x}_*$ is then obtained as $f^{-1}(\mathbf{x}'_{\min})$, and Algorithm 1 is invoked to find its correspondence $\mathbf{x}'_*$. To calculate the inverse APAP warp $f^{-1}(\mathbf{x}'_{\min})$, we find the nearest neighbor of $\mathbf{x}'_{\min}$ in $\{f(\mathbf{x}_i)\}_{i=1}^N$, then warp $\mathbf{x}'_{\min}$ to I using the inverse $\mathbf{H}^{-1}(\mathbf{x})$ of the input-dependent homography (3) of the nearest neighbor point.

The newly inserted correspondence $\{\mathbf{x}_*, \mathbf{x}'_*\}$ is appended to $\mathcal{X}$, if $E(\mathbf{x}'_*)$ is less than ω (default ω is 1000).

Else, the new correspondence is considered unsatisfactory and discarded. In any case, $\mathbf{x}'_{\min}$ is appended to $\mathcal{L}$ to prevent it from being selected again in the next iteration. The above warp adaptation process is repeated until the overlap area is sufficiently covered by feature correspondences.

As an indication of runtime, invoking Algorithm 2 on the image pair in Fig. 5 inserted 81 new correspondences in 65 seconds, among which 11 correspondences were accepted.

4. Results

Evaluation of flow-based methods We used the *truck* image pair from [24]; see Fig. 4. Three state-of-the-art flow-based dense and semi-dense correspondence methods were evaluated [14, 15, 7]. Before obtaining the dense correspondences, we pre-warped one of the images using a homography estimated from sparse SIFT keypoint matches. This served to simplify the problem for the flow-based methods. Further, RANSAC was invoked with a tight inlier threshold (1 pixel) to ensure high-quality correspondences, before APAP warp [23] (the baseline) was estimated.

Despite the above precautions, the stitching results in Fig. 4 exhibit significant local distortions. This indicates that many of the correspondences are actually inaccurate. The small error tolerance of RANSAC still allowed sufficient local deviations (e.g., due to repetitive textures) that distorted the warp; see the supplementary material for the actual data used. While such local inaccuracies may not

Figure 4. APAP stitching results using dense correspondences from three flow-based methods. Since optical flow [14] and SIFT flow [15] produce very dense flow fields, to avoid excessive clutter, we display only 1000 randomly sampled correspondences.

affect motion analysis or segmentation, they are fatal for accurate image stitching using spatially varying warps.

Comparisons with state-of-the-art stitching methods
We compared our method (abbreviated as APAP+CI)[1] against other state-of-the-art approaches, namely the original APAP method [23] and parallax-tolerant image stitching [24]. We used publicly available images by Zaragoza et al. and Zhang and Liu, as well as additional images collected by us. Due to space limitations, only a few results can be shown here; see supplementary material for more results.

For APAP warps, we used the code shared by Zaragoza et al. For parallax-tolerant image stitching, we simply reprinted the results (where available) from the project page of Zhang and Liu. For newly collected images, we executed our own implementation of Zhang and Liu's method.

Parameter settings for our method are as follows: $\sigma = 8$ in (6) , $\mathbb{D}$ in (9) is a 31×31 subwindow, $\epsilon = 100$, $\eta = 0.5$, $\rho = 15$, and $\omega = 1000$ in Algorithm 2.

In image pairs with very serious depth parallax, not all pixels have valid correspondences in the other view. Theoretically, the true warping function must "fold over" or be discontinuous to correctly align the images. Such characteristics are not supported by APAP or the content preserving warp (CPW) [16] used in parallax-tolerant image stitching. Following Zhang and Liu, we thus apply seam cut to composite the images and remove ghosting.

Figs. 5 and 6 show results on two image pairs used by Zhang and Liu. In Fig. 5, parallax-tolerant image stitching produced significant distortions on the glass building. This was likely due to the concentration of the local homography on the major building to the right, and neglecting the

other regions not lying on the same plane (cf. Fig. 2). In contrast, APAP warp was more capable of globally aligning the images; notice that the glass building was not distorted. However, unpleasant distortions exist around the smokestack - due to a lack of feature matches in this region, the warp was "dragged away" by existing feature matches on the lower building. Our method APAP+CI rectified the distortion by inserting new correspondences in the appropriate positions. In Fig. 6, observe the distortions on the pavilion produced by parallax-tolerant image stitching. Overall, APAP warp accurately aligned the whole image, however, due to the lack of feature correspondences, the tower in the background appeared discontinuous. This was rectified by APAP+CI with the insertion of new correspondences.

Similar results on two more challenging image pairs are shown in Figs. 7 and 8; these are newly collected data. Our results show that by inserting new correspondences to adapt the warp, our method rectifies the weakness of APAP warp.

5. Conclusions

Wide-baseline image stitching is a challenging problem. Flow-based methods often fail to produce dense *and* accurate correspondences, while spatially varying warps are only flexible up to the sparse set of keypoint matches given. We presented a novel data-driven warp adaption scheme for APAP image stitching. A core step in our algorithm is a correspondence insertion technique. Our method improves upon the original APAP warp, which fails when the overlap region is correspondence-poor. Our results also show that it is crucial to accurately align the images throughout the overlap area, even if sophisticated compositing is used.

---

[1] Source code will be made available on our homepages.

Figure 5. Comparing three methods on *truck* image pair. Inserted correspondences by APAP+CI are shown as yellow points.
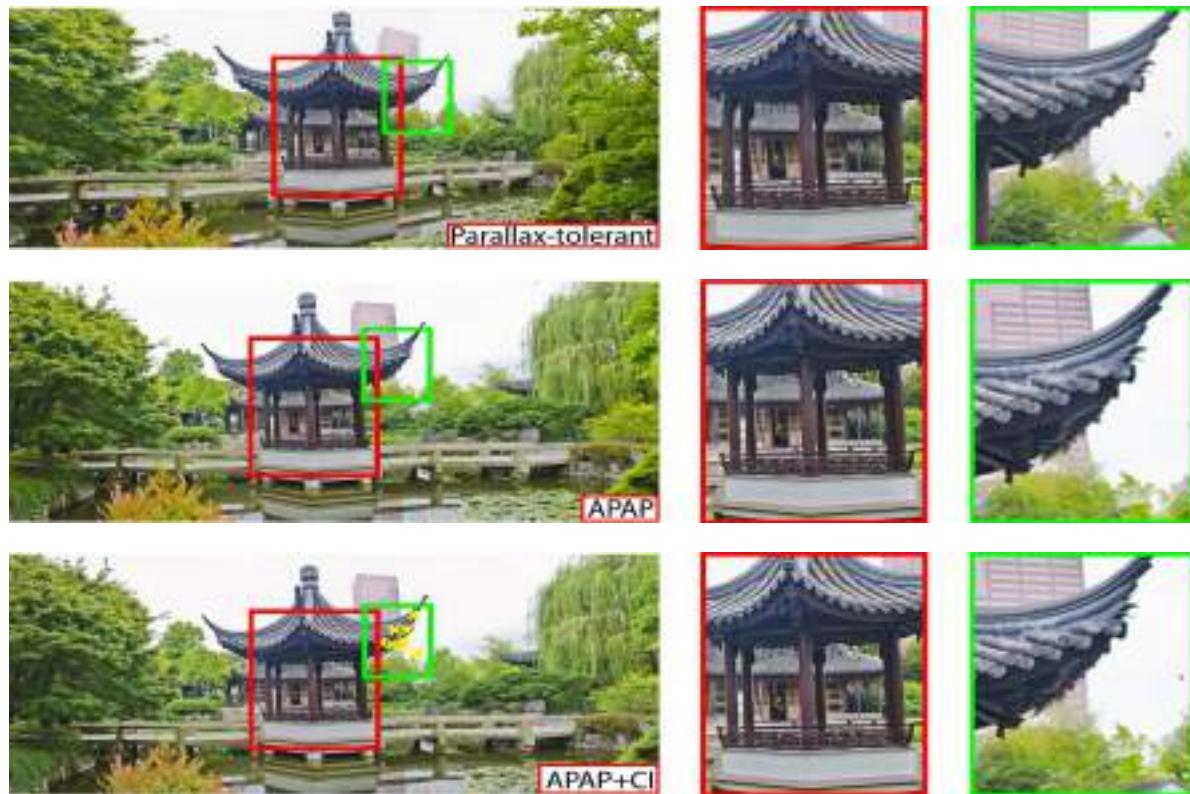

Figure 6. Comparing three methods on *temple* image pair. Inserted correspondences by APAP+CI are shown as yellow points.

Figure 7. Comparing three methods on *shopfront* image pair. Inserted correspondences by APAP+CI are shown as yellow points.
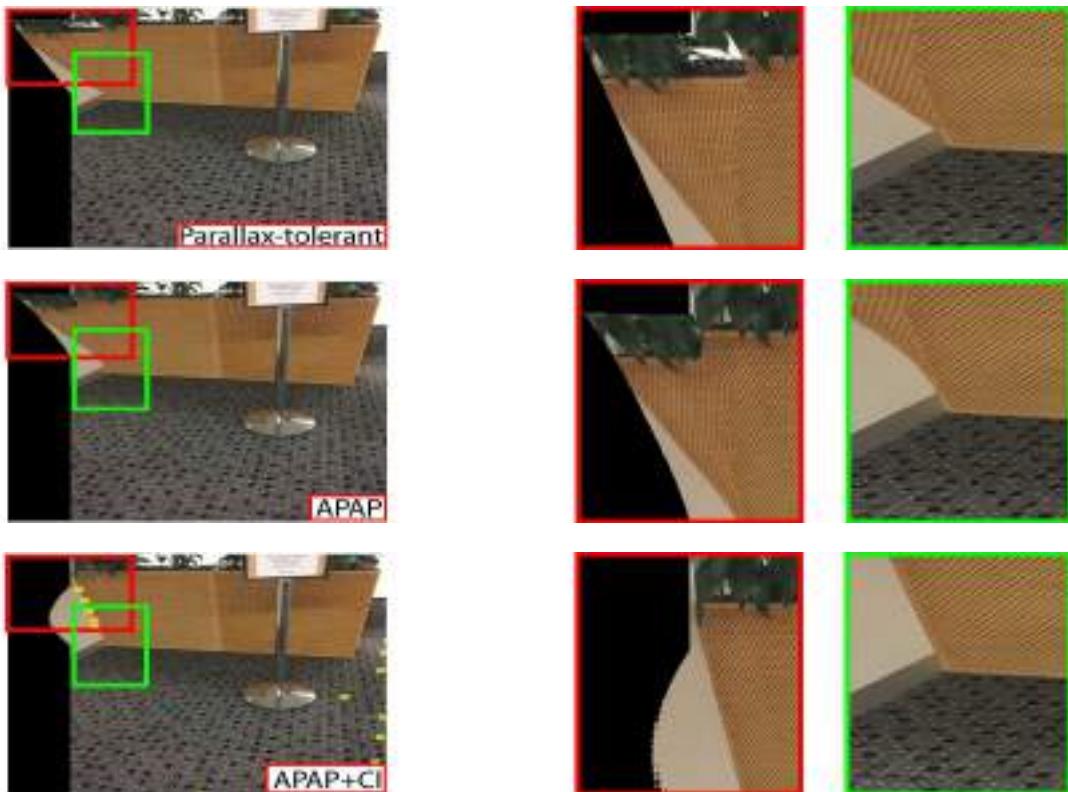

Figure 8. Comparing three methods on *lobby* image pair. Inserted correspondences by APAP+CI are shown as yellow points.

References

[1] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photo-montage. In *ACM SIGGRAPH*, 2004. 2, 4, 5, 10

[2] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Computing and rendering point set surfaces. *IEEE TVCG*, 9(1):3–15, 2003. 4

[3] S. Baker and I. Matthews. Lucas-Kanade 20 years on: a unifying framework. *IJCV*, 56(3):221–255, 2004. 2, 3, 4

[4] A. Bartoli, M. Perriollat, and S. Chambon. Generalized thin-plate spline warps. *IJCV*, 88(1):85–110, 2010. 2

[5] A. Bartoli and A. Zisserman. Direct estimation of non-rigid registration. In *BMVC*, 2004. 2

[6] M. Brown and D. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 74(1):59–73, 2007. 1

[7] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(3):500–513, Mar. 2011. 2, 5, 6, 10, 12, 13, 14, 15

[8] T.-J. Cham and R. Cipolla. Automated B-spline curve representation incorporating MDL and error-minimizing control point insertion strategies. *IEEE TPAMI*, 21(1):49–53, 1999. 2

[9] C.-H. Chang, Y. Sato, and Y.-Y. Chuang. Shape-preserving half-projective warps for image stitching. In *CVPR*, 2014. 1

[10] J. Gao, S. J. Kim, and M. S. Brown. Constructing image panoramas using dual-homography warping. In *CVPR*, 2011. 1

[11] J. Harel, C. Koch, and P. Perona. Graph-Based Visual Saliency. In *NIPS*, 2006. 4

[12] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2nd edition, 2008. 2

[13] J. Jia and C.-K. Tang. Image stitching using structure deformation. *IEEE TPAMI*, 30(4):617–631, 2008. 2

[14] C. Liu, W. Freeman, and E. Adelson. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009. 2, 5, 6, 10, 12, 13, 14, 15

[15] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):978–994, May 2011. 2, 5, 6, 10, 12, 13, 14, 15

[16] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3d video stabilization. In *ACM SIGGRAPH*, 2009. 1, 6

[17] J. R. Magnus. On differentiating eigenvalues and eigenvectors. *Econometric Theory*, 1(2):179–191, 1985. 4

[18] S. Marsland and C. J. Twining. Constructing data-driven optimal representations for iterative pairwise non-rigid registration. In *Second International Workshop on Biometric Image Registration*, 2003. 2

[19] H.-Y. Shum and R. Szeliski. Construction of panoramic mosaics with global and local alignment. *IJCV*, 36(2), 2000. 2

[20] R. Szeliski. Image alignment and stitching: a tutorial. TechReport MSR-TR-2004-92, Microsoft Research, 2004. 1

[21] R. Szeliski and J. Coughlan. Spline-based image registration. *IJCV*, 22(3):199–218, 1997. 2

[22] W. Xu and J. Mulligan. Performance evaluation of color correction approaches for multi-view image and video stitching. In *CVPR*, 2010. 4

[23] J. Zaragoza, T.-J. Chin, M. S. Brown, and D. Suter. As-projective-as-possible image stitching with Moving DLT. In *CVPR*, 2013. 1, 2, 3, 4, 5, 6, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20

[24] F. Zhang and F. Liu. Parallax-tolerant image stitching. In *CVPR*, 2014. 2, 5, 6, 10, 11, 16, 17

Supplementary Material for
Correspondence Insertion for As-Projective-As-Possible Image Stitching

1. Supplementary material overview

This supplementary material provides results on additional image pairs. In particular, Sec. 2 shows stitching results using dense correspondences from three flow-based methods [7, 14, 15], and Secs. 3 and 4 compare our novel image stitching method (APAP+CI) against other state-of-the-art methods.

2. Evaluation of dense correspondences from flow-based methods

Here, we evaluate the quality of the dense correspondences from flow-based methods for image stitching. The same settings from the main paper (Section 4) were applied here, i.e.,

1. One of the images from each pair were pre-warped using a homography estimated from sparse SIFT keypoint matches; see Figs. 9(a), 10(a), 11(a) and 12(a). This served to simplify the data for the flow-based methods.
2. A flow-based method [7, 14, 15] was invoked to obtain dense correspondences.
3. RANSAC was run with a tight inlier threshold (1 pixel) to validate the correspondences. Typically, even after RANSAC, a very large number of correspondences remained, e.g., on the *truck* datatset, the optic flow implementation of [14] found 195868 correspondences, and SIFT Flow [15] found 403308 correspondences.
4. The APAP warp [23] was used as a baseline method to stitch each image pair using the dense correspondences.
5. Seam cut pixel selection [1] is applied after alignment to remove ghosting and alignment errors.

Figs. 9, 10, 11 and 12 show the results.

In Fig. 9, all three methods produce significant distortions on the chimney and the truck, despite the dense correspondences. In Fig. 10, dense correspondences were obtained around the pavilion. However, in Fig. 10(b) the pavilion is seriously distorted, and in Figs. 10(d) and 10(f) the two separate pillars of the pavilion are merged into one and the tower in the background also appears discontinuous. This points to the inaccuracies in the dense correspondences. In Fig. 11, the inaccuracies in the dense correspondences are obvious, especially around top left corner of Fig. 11(e). This leads to discontinuities in the balcony and missing pillars.

The scene in Fig. 12 contains two apparent planes, and SIFT was able to find good sparse correspondences from only one of them (the floor). Thus, the prewarping result using a homography cannot wholly align the images well. Inevitably, this causes problems for the flow-based methods. Large displacement optical flow [7] found a large amount of correspondences on the ground; however, the region on the wall of the flower bed was not covered. The optical flow implementation of [14] and SIFT Flow [15] captured matches on the wall, but still produced stitching results with significant artifacts in Figs. 12(d) and 12(f). This is due to the repeated textures on the wall which lead to inaccurate dense correspondences.

3. Further comparisons on image pairs with significant depth parallax

In this section, results are shown using images of a scene with significant depth parallax. We compare our method (abbreviated as APAP+CI) with the state-of-the-art methods, namely the baseline APAP method [23] and the parallax-tolerant image stitching [24] in Figs. 13 and 14. Similar to the settings in the main paper (Section 4), here, we apply seam cut pixel selection [1] to remove ghosting or alignment errors in the stitched results.

In Fig. 13(b), parallax-tolerant image stitching introduces notable visual artifacts (green window). In Fig. 13(c), APAP produces significant distortion (red window), which is rectified by APAP+CI with the insertion of new correspondences.

For the images shown in Fig. 14(a), there are two apparent planes; however most of the feature matches are on one plane only (the wall). Seam cut produced unexpected artifacts in Figs. 14(b) and 14(c). By inserting new correspondences, our method is able to provide a significantly better global alignment.

4. Comparisons on image pairs without significant parallax

In this section, we present results on the additional three images pairs that have taken of scenes without significant parallax. Because parallax is not present, all results are generated without using seam cut blending. Seam cut, however, is an important step in parallax-tolerant image stitching [24], so we only compare our method (APAP+CI) with results obtained from a single homography (baseline) and the APAP method. Since seam cut blending is not used, we run our method on the overlapped image region only, which is slightly different from the Algorithm 2 in the main paper. Figs. 15, 16 and 17 show the results.

The single homography model works under the assumption that the images are sufficiently far away or taken with a camera undergoing pure rotational motion. For these images, this imaging condition is not satisfied and a single homography is not sufficient to align the images. As shown in Figs. 15(b), 16(b) and 17(b), warping with a single homography introduces significant ghosting artifacts in the stitching results. APAP warp is able to provide a more accurate alignment, but fail if there are insufficient point matches (e.g. pillar in Fig. 15(c), arch in Fig 16(c), and railing and eave in Fig 17(c)). Our method automatically adds new correspondences and rectifies the weakness of the APAP warp producing results that have a better overall alignment.

(a) Pre-warp input images using a homography estimated from SIFT keypoint matches.
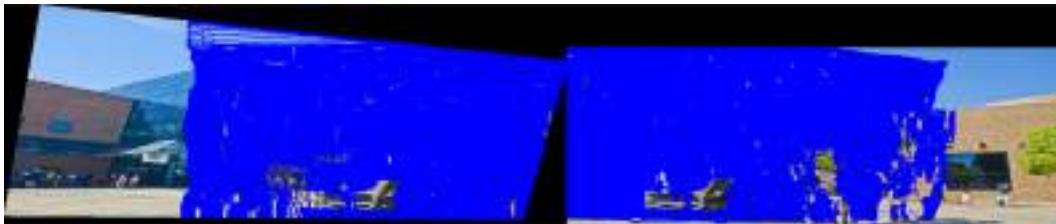
(b) Image stitching result using APAP warp [23] estimated from a set of semi-dense correspondences (validated by RANSAC and shown as red points in the final stitched image) produced by Large Displacement Optical Flow [7].

(c) The optic flow implementation of [14] produced 195868 correspondences (after RANSAC validation).

(d) Image stitching result using APAP warp [23] estimated from the correspondences in (c).


(e) SIFT Flow [15] produced 403308 correspondences (after RANSAC validation).

(f) Image stitching result using APAP warp [23] estimated from the correspondences in (e).

Figure 9. Dense correspondences and stitching results of three flow-based methods on the *truck* image pair.

(a) Pre-warp input images using a homography estimated from SIFT keypoint matches.
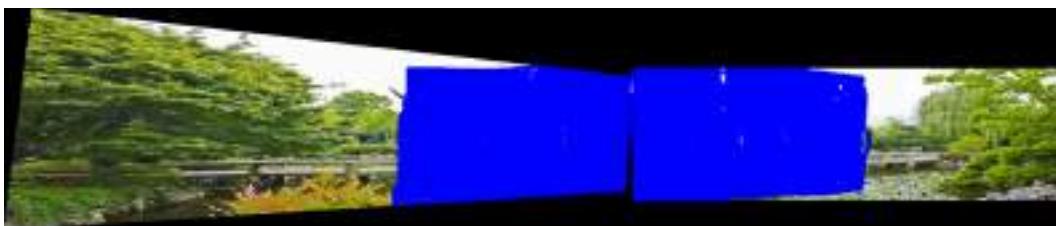
(b) Image stitching result using APAP warp [23] estimated from a set of semi-dense correspondences (validated by RANSAC and shown as red points in the final stitched image) produced by Large Displacement Optical Flow [7].


(c) The optic flow implementation of [14] produced 147858 correspondences (after RANSAC validation).

(d) Image stitching result using APAP warp [23] estimated from the correspondences in (c).

(e) SIFT Flow [15] produced 317753 correspondences (after RANSAC validation).
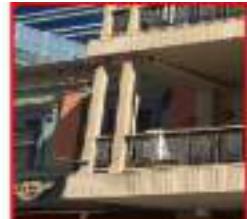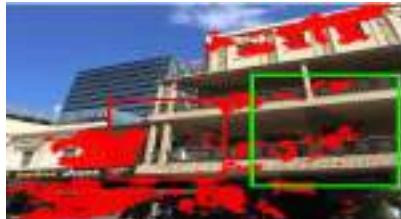
(f) Image stitching result using APAP warp [23] estimated from the correspondences in (e).

Figure 10. Dense correspondences and stitching results of three flow-based methods on the *temple* image pair.

(a) Pre-warp input images using a homography estimated from SIFT keypoint matches.

(b) Image stitching result using APAP warp [23] estimated from a set of semi-dense correspondences (validated by RANSAC and shown as red points in the final stitched image) produced by Large Displacement Optical Flow [7].
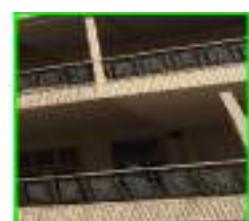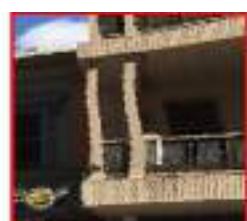
(c) The optic flow implementation of [14] produced 83318 correspondences (after RANSAC validation).

(d) Image stitching result using APAP warp [23] estimated from the correspondences in (c).

(e) SIFT Flow [15] produced 165789 correspondences (after RANSAC validation).
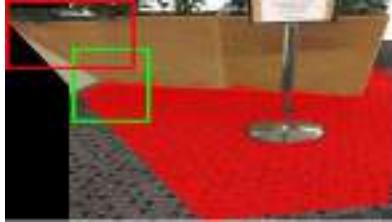
(f) Image stitching result using APAP warp [23] estimated from the correspondences in (e).
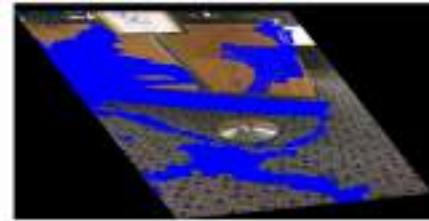
Figure 11. Dense correspondences and stitching results of three flow-based methods on the *shop front* image pair.
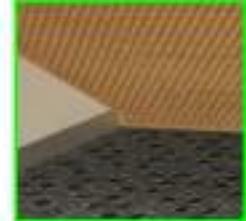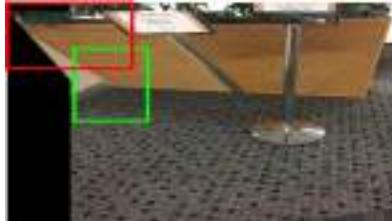
(a) Pre-warp input images using a homography estimated from SIFT keypoint matches.
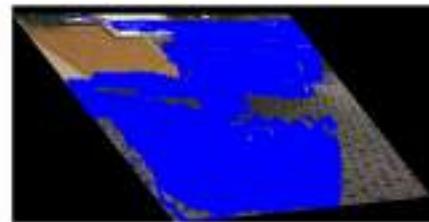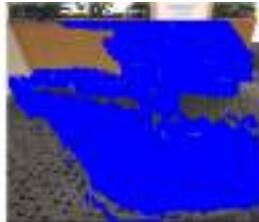
(b) Image stitching result using APAP warp [23] estimated from a set of semi-dense correspondences (validated by RANSAC and shown as red points in the final stitched image) produced by Large Displacement Optical Flow [7].
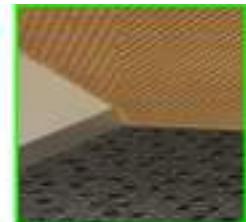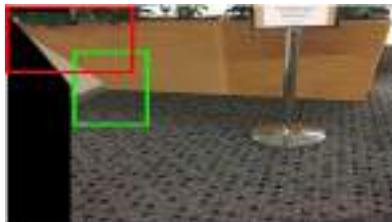
(c) The optic flow implementation of [14] produced 118823 correspondences (after RANSAC validation).

(d) Image stitching result using APAP warp [23] estimated from the correspondences in (c).

(e) SIFT Flow [15] produced 241276 correspondences (after RANSAC validation).
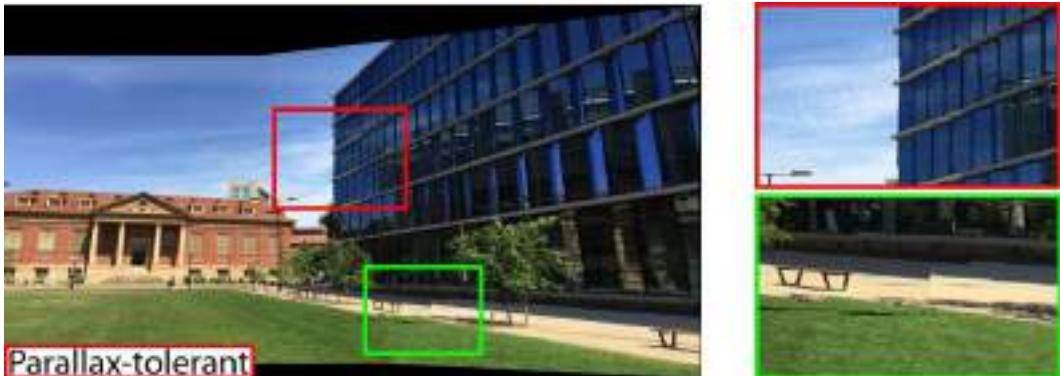
(f) Image stitching result using APAP warp [23] estimated from the correspondences in (e).
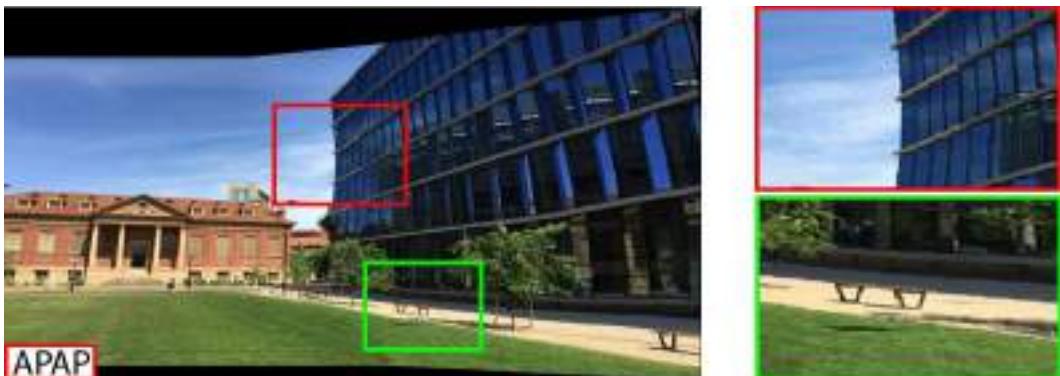
Figure 12. Dense correspondences and stitching results of three flow-based methods on the *lobby* image pair.
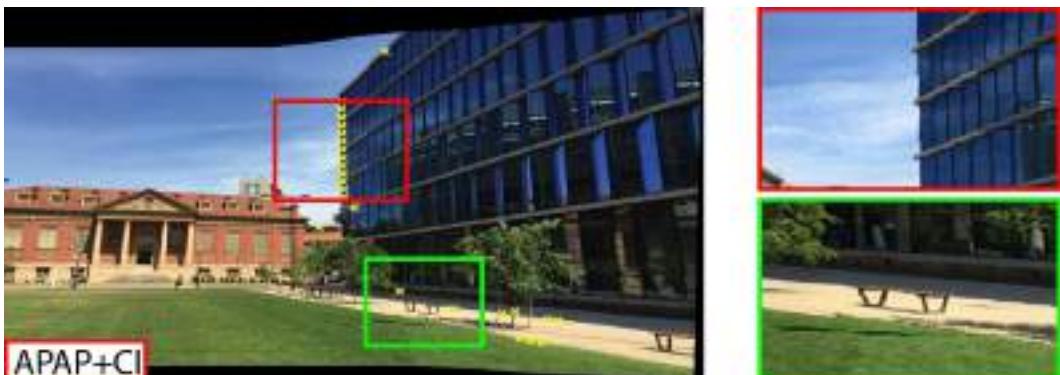
(a) Pre-warp input images using a homography estimated from SIFT keypoint matches.

(b) Result using parallax-tolerant image stitching. [24].

(c) Result using APAP warp [23].

(d) Result after adding 25 new correspondences using the proposed method.

Figure 13. This figure shows a comparison of three methods on the *lawn* image pair. Inserted correspondences by APAP+CI are shown as yellow points.

(a) Input images.

(b) Result using parallax-tolerant image stitching [24].

(c) Result using APAP warp [23].
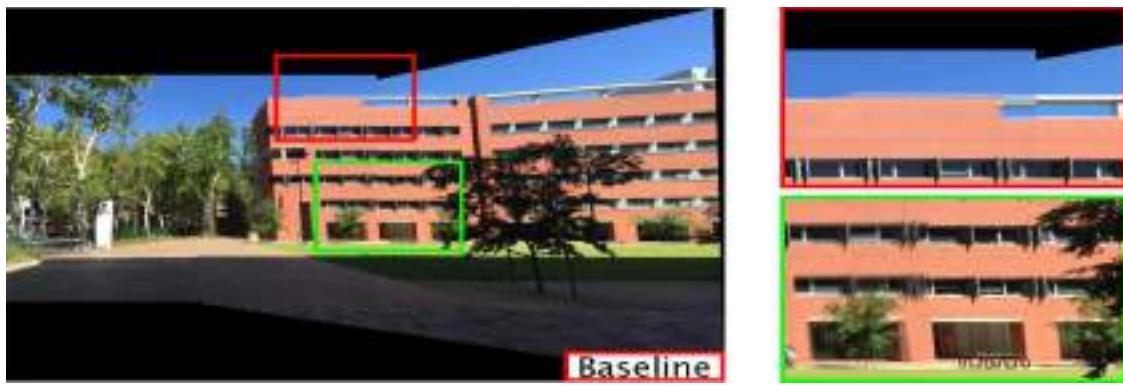
(d) Image stitching result after adding 23 new correspondences using the proposed method.
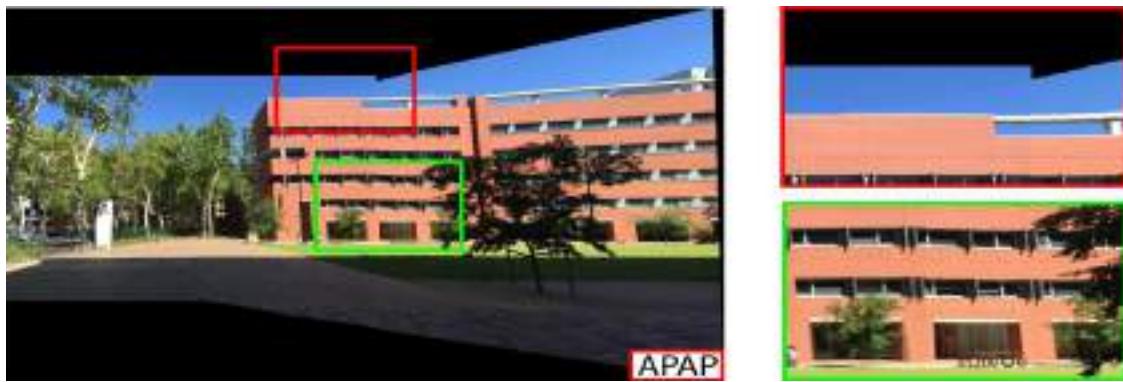
Figure 14. Comparing of three methods on the *break room* image pair. Inserted correspondences by APAP+CI are shown as yellow points.
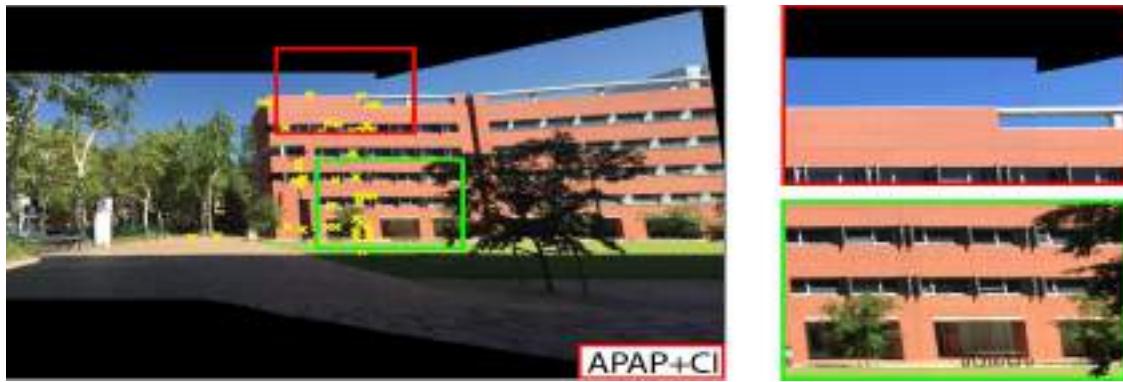
(a) Input images.

(b) Result using a single homography.

(c) Result using APAP warp [23].

(d) Image stitching result after adding 38 new correspondences using the proposed method.
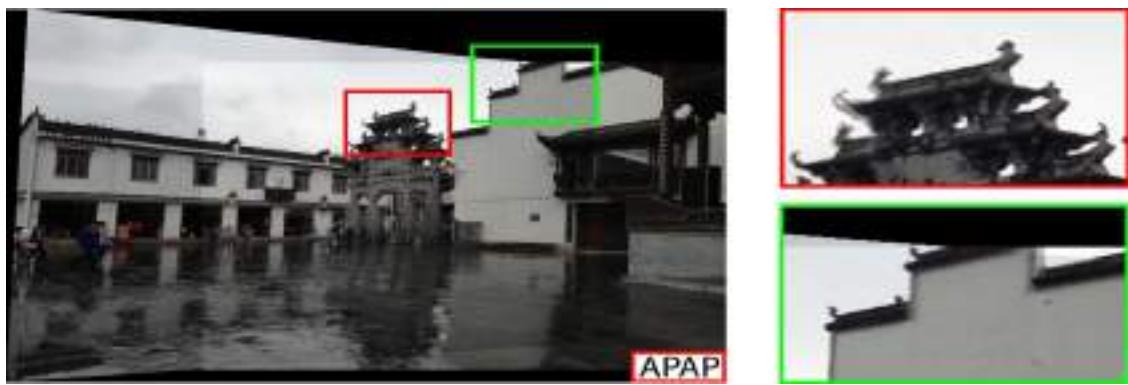
Figure 15. This figure shows a comparison of three methods on the *building* image pair. Inserted correspondences by APAP+CI are shown as yellow points.

(a) Input images.

(b) Result using a single homography.

(c) Result using APAP warp [23].
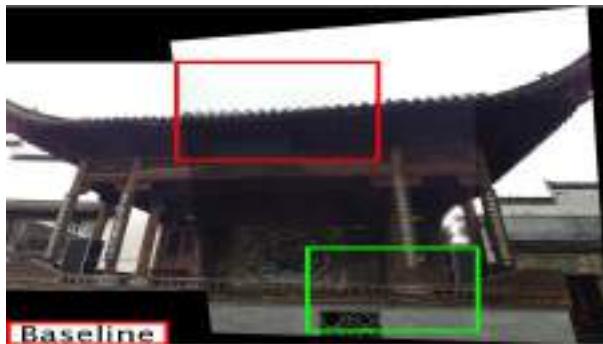
(d) Image stitching result after adding 6 new correspondences using the proposed method.
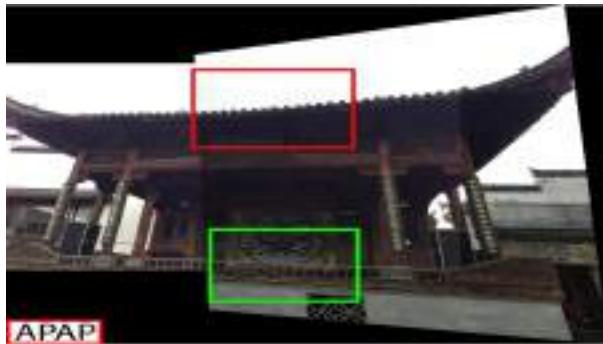
Figure 16. This figure shows a comparison of three methods on the *arch* image pair. Inserted correspondences by APAP+CI are shown as yellow points.
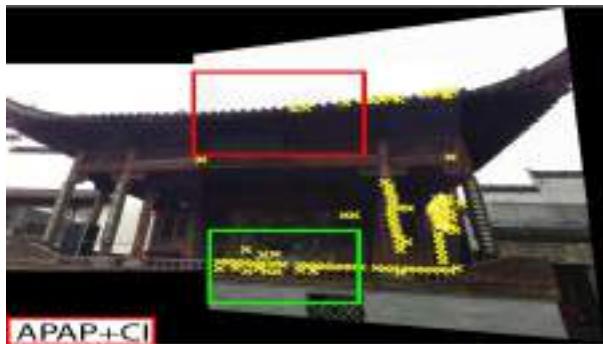
(a) Input images.

(b) Result using single homography.

(c) Result using APAP warp [23].

(d) Image stitching result after adding 93 new correspondences using proposed the method.

Figure 17. This figure shows a comparison of three methods on the *stage* image pair. Inserted correspondences by APAP+CI are shown as yellow points.