April 2012

UILU-ENG-12-2202
DC-255

Holistic Video Stitching for Street Panorama

Zihan Zhou, Kerui Min, and Yi Ma

Coordinated Science Laboratory
1308 West Main Street, Urbana, IL 61801
University of Illinois at Urbana-Champaign

REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comment regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services. Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
April 2012

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
Holistic Video Stitching for Street Panorama

5. FUNDING NUMBERS
NSF IIS 11-16012

6. AUTHOR(S)
Zihan Zhou, Kerui Min, and Yi Ma

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
1308 West Main Street
Urbana, Illinois 61801-2307

8. PERFORMING RGANIZATION REPORT NUMBER
UILU-ENG-12-2202
DC-255

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Science Foundation
4201 Wilson Blvd, Arlington, VA 22203

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official position, policy, or decision, unless so designated by other documentation

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
In this paper, we address how to automatically generate a panorama for a street view from a long video sequence. We model the panorama as a low-rank matrix and formulate the problem as one of robust recovery of the low-rank matrix from highly incomplete, corrupted, deformed measurements (the video frames). We leverage powerful high-dimensional convex optimization tools from compressive sensing of sparse signals and low-rank matrices to solve this problem. In particular, we show how the new method can effectively remove severe occlusions or corruptions (caused by trees, cars, or reflections, etc.), and obtain clean, intrinsic street panoramas that are consistent with all frames. We also show how our method can automatically and robustly establish pixel-wise accurate registration among all the video frames. We demonstrate the effectiveness of our method by conducting extensive experimental comparison with other popular video stitching methods such as AutoStitch and Adobe Photoshop.

14. SUBJECT TERMS
Panorama, street-view, low-rank matrix

15. NUMBER OF PAGES
15

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL

Holistic Video Stitching for Street Panorama

Zihan Zhou[1], Kerui Min[1], and Yi Ma[1,2]

[1]ECE Department, University of Illinois at Urbana-Champaign
[2]Visual Computing Group, Microsoft Research Asia
{zzhou7, kmin12}@illinois.edu, mayi@microsoft.com

Abstract. In this paper, we address how to automatically generate a panorama for a street view from a long video sequence. We model the panorama as a low-rank matrix and formulate the problem as one of robust recovery of the low-rank matrix from highly incomplete, corrupted, deformed measurements (the video frames). We leverage powerful high-dimensional convex optimization tools from compressive sensing of sparse signals and low-rank matrices to solve this problem. In particular, we show how the new method can effectively remove severe occlusions or corruptions (caused by trees, cars, or reflections etc.), and obtain clean intrinsic street panoramas that are consistent with all frames. We also show how our method can automatically and robustly establish pixel-wise accurate registration among all the video frames. We demonstrate the effectiveness of our method by conducting extensive experimental comparison with other popular video stitching methods such as AutoStitch and Adobe Photoshop.

1 Introduction

Recently, driven by industrial applications such as map building, virtual reality, and automatic navigation in urban environments, there has been tremendous interest and effort for building large-scale structure and texture models for urban areas from *street view videos*, which are taken by a moving camera (mounted on cars) that capture side views of the streets (see Figure 1). One of the fundamental problems in compressing and processing such video data is to align and stitch the video frames together to generate a seamless panorama of the streets.

In the image stitching literature, a planar projective deformation model is one of the most popular parametric models, and consists of estimating a 3×3 homography matrix between two images to be aligned. A limitation of this model is that it assumes all the scene structures lying on a plane or the center of projection being fixed, which is often violated in practice. Fortunately, in street view videos, there often exists a dominant plane in the scene, which corresponds to the consecutive building facades on one side of the street. Moreover, in many real applications, one may be particularly interested in accurate recovery of the structure and appearance of this dominant plane only, as it often suffices to produce compact yet visually pleasing large-scale city street views. Indeed, commercial products such as Google Map Street View and Bing Maps all build upon this

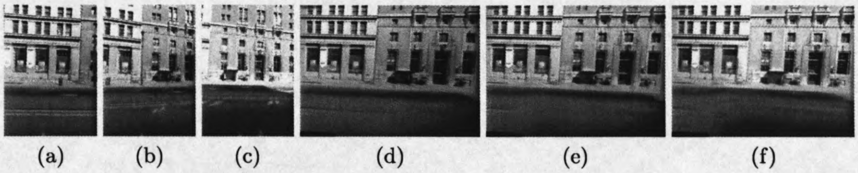


Fig. 1. Intrinsic views with different exposures. **(a)-(c):** Sample input frames from Google Map Street View database. **(d)-(f):** Corresponding intrinsic views recovered by our method after stitching all the frames together. Notice that we recover one panorama for each frame with the same exposure.

model. Therefore, in this work, we focus on how to automatically obtain consistent appearance of the dominant scene plane from street view videos.

Given a set of well aligned input images, conventional image stitching methods use some blending algorithms to compensate for variations such as exposure differences and generate a *single* mosaic image. However, it is known to be difficult for any blending algorithm to strike a good balance between smoothing out low-frequency exposure variations and retaining sharp details in the image. This motivates us to reconsider the following question: Is it really necessary to blend all the input images into a single mosaic image? In fact, each input image can be viewed as an incomplete (windowed) view of a scene plane (building facades along the street) under certain unknown exposure. The key observation here is that if we have multiple complete views of the plane under varying exposures, then they all lie in *a low-dimensional subspace*. For example, under the well known Bias and Gain model [9], the dimension of the subspace is at most two. We call each element within this subspace an *intrinsic view* of the scene. Consequently, instead of blending all the input images into a single mosaic image, we argue in this paper that it is more appropriate to directly recover the complete intrinsic views, or equivalently the low-dimensional subspace, from all the input images in a holistic fashion. Figure 1 shows an example of our results.

However, real images often deviate from the ideal low-dimensional subspace model due to the existence of some parallax, reflection on glasses, and occluding objects in the scene, as shown in Figure 2(a) and (b). They all introduce gross outliers to the imagery data for the blending problem and should be eliminated in the final panorama. In this paper, given the aligned input images, we formulate the robust intrinsic view recovery task as a matrix rank minimization problem, and demonstrate how it can be converted into a convex optimization problem and solved efficiently using recently developed efficient and scalable first-order methods. Somewhat surprisingly, we show that, under this general low-rank subspace assumption, it is possible to recover all missing parts of each input image despite all the aforementioned types of outliers, yielding a complete intrinsic panorama view of the scene for each frame under the same exposure!

In fact, the flexibility of the low-dimensional subspace model goes beyond merely handling the exposure differences. For instance, as one can see in Figure 2, although the building facades are roughly planar, the small structures on the facades, such as windows and doors, do have some depth variation, which

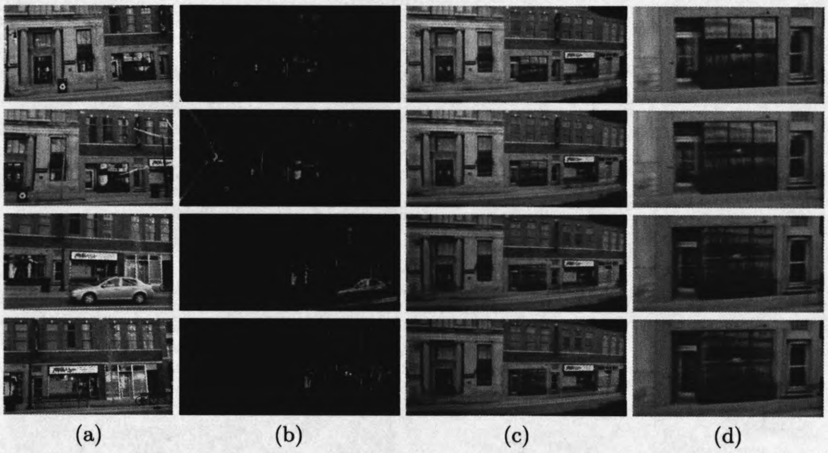


Fig. 2. Intrinsic panoramas from a street video sequence. **(a):** Input frames. **(b):** Residual images. **(c):** Recovered intrinsic panoramas. **(d):** Details of a small region in (c).

is retained in the panorama associated with each view in our final results (Figure 2(d)). Different from occluding objects, keeping these low-frequency variations actually produce more realistic presentation of the real scene than enforcing everything to lie on a single plane.

Remember that the first step for any accurate image stitching is to obtain pixel-wise precise image alignment. While image-intensity based (direct) methods are often used to align sequential frames in a video, they typically assume the intensity remains constant over time. Recently, using the same idea of matrix rank minimization, [10] proposes a novel method called *Robust Alignment by Sparse and Low-rank decomposition* (RASL) for aligning a batch of linearly correlated images. However, like other direct methods, RASL requires a reasonable initialization. In this paper, we extend RASL to automatically register video sequences by initializing it using a robust feature-based plane detection and tracking algorithm, which is a generalized version of the well-known two-frame RANSAC method for homography estimation. By combining the strengths of both intensity-based and feature-based methods, our new method achieves fully automatic alignment with pixel-wise accuracy.

Related Work. The problem of image alignment has been extensively studied in the literature. Existing methods can be roughly classified into two categories. On one hand, direct alignment methods [13, 12] work on image regions and provide accurate registration using local algorithms, but need a good initialization. On the other hand, feature-based methods rely on detecting and matching a set of feature points, such as corners and SIFT features [5, 3]. They do not require an initialization but are often less accurate. Recent work on video registration has been focused on its efficiency and global optimality [14]. In this work, we take the advantage of both types of methods to achieve robust and accurate registration of video frames.

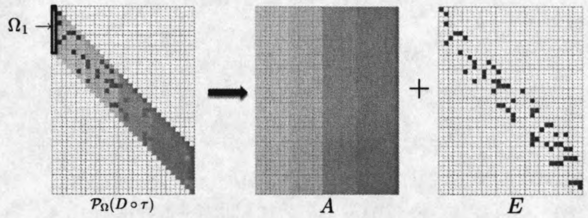


Fig. 3. Illustration of our problem formulation: robust recovery of a low-rank matrix A from highly incomplete measurements within a band along the matrix diagonal.

Given multiple aligned input images, there exist many works addressing the issues in compositing the final mosaic image, such as pixel and seam selection, blending and exposure compensation. Pixel and seam selection techniques aim to eliminate the ghost effects due to moving objects, by using a median filter [7], a minimum likelihood selection criterion [1], or a weighted average in the regions of difference [15]. While some of these methods such as [15] can already compensate for some differences in exposure or color from the source images, more sophisticated blending algorithms have been developed in the literature. [3] uses pyramid blending to compensate for exposure differences, and [11] develops a gradient domain blending method to do seamless object insertion in image editing applications. Several variants of [11] with different cost functions have been studied in [8] to further improve its performance. Meanwhile, readers are referred to [17] for a comprehensive performance evaluation on existing color correction approaches. Finally, [6] proposes to convert each image into a radiance image using its exposure value and then create a stitched, high dynamic range image. While all the aforementioned methods focus on eliminating various types of variations from the input images, it is the novelty of our method to directly model these variations and keep them in the final results.

2 Problem Formulation

We begin introducing our method with a formal definition of our low-dimensional subspace model for video panoramas. Suppose we are given n complete and pixel-wise aligned views (w.r.t. a common coordinate system associated with the dominant plane) of a scene under different exposures. We stack the m pixels of each video frame as a vector, and denote them as $I_1^0, \ldots, I_n^0 \in \mathbb{R}^m$, one for each of the n frames in the sequence. We put these vectors as columns of the matrix

$$A \doteq [I_1^0, \ldots, I_n^0] \in \mathbb{R}^{m \times n}. \tag{1}$$

Then the columns should be highly linearly correlated and the matrix has a very low rank, as illustrated in Figure 3. Notice that ideally, the only difference in the different columns is a scaling factor which corresponds to the global illumination change across views. We call these columns as "intrinsic images."

In an image stitching problem, in each input image I_j we only see a deformed version of a very small portion of the entire scene. In particular, a video sequence along a street can be thought as a sliding window through which each frame sees a small chunk of the street from a different view. If the camera is a perspective projection, then there exist homography matrices $\tau_1, \ldots, \tau_n \in \mathbb{GL}(3)$ which transform the input video frames $I_1, \ldots, I_n$ into a common coordinate system on the dominant plane, respectively.

In addition, each image I_j $(1 \leq j \leq n)$ has a limited field of view and only sees a very small portion of the scene. Hence, there is an associated support Ω_j that indicates the observable region (entries) from the j-th view, as illustrated in Figure 3. We write $\mathcal{P}_{\Omega_j}(I_j)$ as the projection of I_j to the space of vectors supported on Ω_j. With a slight abuse of notation, we also use $\Omega_j \in \mathbb{R}^m$ as a vector to represent the observed pixels in I_j^0, where $\Omega_j(k) = 1$ if the k-th pixel is observable in I_j^0, and $\Omega_j(k) = 0$ otherwise. Hence the video frames are related to the intrinsic images as

$$\mathcal{P}_{\Omega_j}(I_j \circ \tau_j) = \mathcal{P}_{\Omega_j}(I_j^0). \tag{2}$$

Given the transformation matrices $\tau = \{\tau_j\}_{j=1}^n$, we can write the aligned data matrix as $\mathcal{P}_\Omega(D \circ \tau) \doteq [\mathcal{P}_{\Omega_1}(I_1 \circ \tau_1), \ldots, \mathcal{P}_{\Omega_n}(I_n \circ \tau_n)]$, where $\Omega \doteq [\Omega_1, \ldots, \Omega_n]$ are the supports associated with all the views. Then the image stitching problem naturally reduces to the following low-rank matrix completion problem:

$$\min_{\tau, A} \operatorname{rank}(A) \quad \text{s.t.} \quad \mathcal{P}_\Omega(D \circ \tau) = \mathcal{P}_\Omega(A). \tag{3}$$

In practice, the low-rank structure of the aligned images can be easily violated, due to the presence of parallax, reflections and moving objects in the scene. Since these errors typically affect only a small fraction of all pixels in an image, we can model them as sparse errors whose nonzero entries can have large magnitude. Let $\boldsymbol{e}_j$ represent the error corresponding to the j-th frame: $I_j \circ \tau_j = I_j^0 + \boldsymbol{e}_j$. Let $E = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]$ be the matrix with all the error vectors as columns. Then to recover the low-rank intrinsic images I_j^0, we actually need to solve the following more challenging problem of recovering a low-rank matrix from highly incomplete *and* corrupted observations:

$$\min_{\tau, A, E} \operatorname{rank}(A) + \nu \|E\|_0 \quad \text{s.t.} \quad \mathcal{P}_\Omega(D \circ \tau) = \mathcal{P}_\Omega(A + E), \tag{4}$$

where the ℓ_0-norm $\|\cdot\|_0$ counts the number of nonzero entries of a matrix, and $\nu > 0$ is a parameter that trades off the rank of the solution versus the sparsity of the error.

To summarize, our goal is to recover a set of homographies $\tau_1, \ldots, \tau_n$ that align all the frames to a common world plane coordinate as well as the corresponding complete intrinsic views, by minimizing the rank of a matrix A which agrees with the aligned input images $\{I_j\}_{j=1}^n$ on the observed regions Ω, up to some sparse gross errors E. Notice that here (τ, A, E) are all unknowns.

The rest of the paper is organized as follows. In Section 3, by assuming that the correct homographies τ are given, we introduce an efficient and effective solution to (4) via convex programming. Then, to obtain the homographies for a video sequence, we develop a new robust video frame alignment algorithm in Section 4. We conduct extensive experiments to illustrate the performance of our method and compare with other state of the art techniques in Section 5.

3 Robust Low-rank Panoramas via Convex Optimization

In this section, we show how to solve the problem (4) when the correct transformations τ are given. Note that even with τ given, the objective function of problem (4) is still highly combinatorial, which is in general NP-hard if we are looking for the global optimal solution. However, by the recent advances in convex optimization, we can replace the non-linear functions rank$(\cdot)$ and ℓ_0-norm by their corresponding convex surrogates, as proposed by the work of *Principal Component Pursuit* [4] for solving the robust PCA problem. Specifically, we replace rank$(\cdot)$ by the nuclear norm $\|\cdot\|_*$[1], and $\|\cdot\|_0$ by the $\|\cdot\|_1$ norm[2], which leads to the following convex optimization problem:

$$\min_{A,E} \|A\|_* + \lambda\|E\|_1 \quad \text{s.t} \quad \mathcal{P}_\Omega(D \circ \tau) = \mathcal{P}_\Omega(A + E). \tag{5}$$

In the literature of compressive sensing and low-rank matrix recovery, there have been extensive theoretical results that provide evidences for the effectiveness of using such convex surrogates for recovering sparse signals and low-rank matrices. In particular, [4] has shown that in the case when Ω contains a small constant fraction of the entries, the above convex program succeeds with high probability in recovering the correct low-rank and sparse components (A_0, E_0) under mild conditions. Similar recoverability results have also been obtained for a more general low-dimensional subspace Ω, known as *Compressive Principal Component Pursuit* [16]. In a nutshell, the results in [16] claim that, if (A_0, E_0) are incoherent, then the recovery from (5) is exact with high probability if

$$\dim(Q) \geq C \cdot \log^2 n \times \text{degrees of freedom}(A_0, E_0),$$

where Q is a randomly chosen observable subspace according to the Haar measure. Curious readers are referred to [16] for the detailed proofs. Notice that if the matrix A has a fixed rank r – which is the case in our setting, then a lower-bound on the number of measurements needed is $O(rn\log^2 n)$ which is only a diminishing fraction of the entries $O(n^2)$ of the matrix as n goes to infinity. This is actually the case when the length of the video sequence grows large. The results of [16] suggest that as long as the resolution of the image frame – the size of the support $|\Omega_1|$ in Figure 3 – grows as $O(r\log^2 n)$, then good recovery of the low-rank intrinsic views and sparse errors is possible.

It is easy to see that problem (5), as a convex optimization program, can be solved by the polynomial-time interior point methods. Although this is conceptually correct, such methods are highly inefficient for our problem here, as we

[1] Sum of all singular values of a matrix.

[2] Sum of absolute values of all entries.

Algorithm 1 First-order Method for Solving (6)

```
while not converged ($j = 1, 2, \ldots$) do
  Initialize $R_1 \leftarrow \Theta_1$, $\xi_1 \leftarrow 1$.
  while not converged ($k = 1, 2, \ldots$) do
    $M_k \leftarrow (1 - \xi_k) R_k + \xi_k \Theta_k$;
    $(U_k, \Sigma_k, V_k) \leftarrow \text{svd}\left(A_j^{(0)} + \mu_j^{-1} \mathcal{P}_\Omega M_k\right)$;
    $A_{k+1} \leftarrow U_k \cdot \text{shrink}\left(\Sigma_k, \mu_j^{-1}\right) \cdot V_k^*$;
    $E_{k+1} \leftarrow \text{shrink}\left(E_j^{(0)} + \mu_j^{-1} \mathcal{P}_\Omega M_k, \lambda \mu_j^{-1}\right)$;
    $\Theta_{k+1} \leftarrow \Theta_k + \frac{\mu_j}{2}(D \circ \tau - \mathcal{P}_\Omega(A_{k+1} + E_{k+1}))$;
    $R_{k+1} \leftarrow (1 - \xi_k) R_k + \xi_k \Theta_{k+1}$;
    $\xi_{k+1} \leftarrow 2/(1 + \sqrt{1 + 4/\xi_k^2})$;
  end while
  (Let $(A^\dagger, E^\dagger)$ be the converged solution);
  $\left(A_{j+1}^{(0)}, E_{j+1}^{(0)}\right) \leftarrow (A^\dagger, E^\dagger)$;
  $\mu_{j+1} \leftarrow \gamma \cdot \mu_j$;
end while
```

are processing video data of high resolutions and long sequences. Therefore, we adopt a first-order method similar to [2], where instead of directly solving the constrained problem (5), we solve the following smoothed Lagrangian problem:

$$\begin{aligned}&\mathcal{L}(A, E, \Theta)\\ &= \|A\|_* + \lambda \|E\|_1 + \frac{\mu}{2}\left(\|A - A^{(0)}\|_F^2 + \|E - E^{(0)}\|_F^2\right) - \langle \Theta, \mathcal{P}_\Omega(A + E - D \circ \tau)\rangle, \quad (6)\end{aligned}$$

where μ is the smoothing parameter, Θ is the Lagrange multiplier matrix, and $A^{(0)}, E^{(0)}$ are two fixed matrices. If $A^{(0)} = A_0$ and $E^{(0)} = E_0$, then we have that the optimal solution of problem (6) is the same as problem (5). Since we do not know A_0, E_0 beforehand, we iteratively update $A^{(0)}, E^{(0)}$ as well as the dual variable Θ one at a time while fixing the others.

The detailed algorithm is summarized in Algorithm 1, which is guaranteed to converge to the optimal solution as proved in [2]. Empirically, fixing $\gamma = 0.9$, it usually takes 40 to 60 iterations to converge.

4 Robust and Accurate Video Registration

We have seen from previous sections that by imposing low-rankness on the desired solution, one can robustly and efficiently recover the complete intrinsic views of the scene despite gross errors. In this section, we show that with some proper modifications, the same idea of matrix rank minimization can be used to obtain accurate estimates of the homography matrices among all video frames and hence the transformations between the image frames and the intrinsic views.

Given n input images $\{I_j\}_{j=1}^n$, recall that the ideal observation model is $\mathcal{P}_{\Omega_j}(I_j \circ \tau_j) = \mathcal{P}_{\Omega_j}(I_j^0)$ for the j-th image. Denote $R = \bigcap_{j=1}^n \Omega_j$ as the intersection of observable regions among images considered – see Figure 4 for an illustration. If R is *not* empty, we write $\mathcal{P}_R(I_j \circ \tau_j), j = 1, \ldots, n$ as the projection

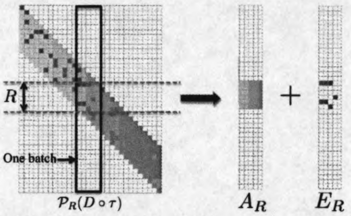


Fig. 4. Illustration of our formulation of the robust image registration problem, restricted to a batch of images considered.

of the aligned input images to the space of vectors supported on R. Then, if we stack each $\mathcal{P}_R(I_j \circ \tau_j)$ as column of a new data matrix:

$$\mathcal{P}_R(D \circ \tau) = [P_R(I_1 \circ \tau_1), \ldots, P_R(I_n \circ \tau_n)], \tag{7}$$

this matrix should also have a very low rank. In the presence of errors, we can write our observation model as:

$$\mathcal{P}_R(D \circ \tau) = A_R + E_R, \tag{8}$$

where A_R, E_R represent the low-rank component and sparse error component, respectively, with their k-th entries being zero for any $k \notin R$. Then, using the same argument as in Section 2, we can cast the problem of joint image alignment as the following optimization problem:

$$\min_{A_R, E_R, \tau} \|A_R\|_* + \lambda \|E_R\|_1 \quad \text{s.t} \quad \mathcal{P}_R(D \circ \tau) = A_R + E_R. \tag{9}$$

In fact, this problem has been extensively studied in the recent work [10], where it is called *Robust Alignment by Sparse and Low-rank decomposition* (RASL). In [10], given a good initialization, (9) is solved by iteratively linearizing the nonlinear equality constraint at the current estimate of τ, yielding a sequence of convex programs whose solutions converge quadratically to the correct alignment. It has been shown in [10] that RASL is able to achieve pixel-wise alignment accuracy over a wide range of realistic misalignments and corruptions.

However, in order to apply RASL to street view video sequences, there are two important issues which need to be addressed. First, as a local method, a good initialization of the transformation parameters is required by RASL. Second, for a typical long video sequence, there is often no common regions among all frames – see Figure 4. In the rest of the section, we show how to resolve these two problems.

4.1 RANSAC-based dominant plane detection and tracking

Since our goal is to compute the homography matrices induced by the dominant scene plane, it can be done efficiently using a RANSAC-type algorithm on feature point trajectories obtained by any standard tracking algorithm. We here discuss such a method in more details.

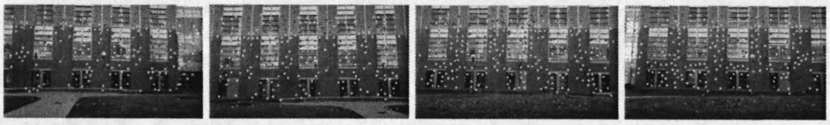


Fig. 5. Plane detection and tracking results using our RANSAC-based algorithm. Green dots: inliers. Red dots: outliers. Notice that our method effectively eliminate as outliers all features that are off the plane or caused by reflections.

In computer vision literature, RANSAC has been the most widely used algorithm for robustly estimating the planar homography between two images. To register video sequence, one may simply apply RANSAC to each pair of adjacent frames and combine the result across multiple pairs. However, the plane model detected in this way may not be consistent over time: in the presence of mis-tracked features, spurious features in each frame, or multiple planes, feature points that fit one homography between a pair of adjacent frames could be outliers for the homography found for the next pair of frames. Therefore, in this work we use a generalized RANSAC algorithm, which *directly samples feature point trajectories*, instead of independently sampling point correspondences between every two frames. Since the method is still based on sampling consensus, it consists of multiple trials of the same procedure followed by a selection of the best result (in terms of the number of inliers) from these trials. The procedure of one trial is as follows.

1. Given an input image sequence, we randomly choose a pair of adjacent frames. A homography matrix is then estimated using four randomly chosen trajectories which overlap with these two frames.
2. Given a fixed threshold ϵ, we classify all the other trajectories that overlap with these two frames into inliers and outliers using the estimated homography matrix. Then, using the inlying trajectories we can estimate additional homography matrices between other adjacent pairs of subsequent frames, which can in turn be used to detect more inlying trajectories. We repeat this step until all the frames are processed.

Note that the only parameter for this scheme is the threshold ϵ. Since our goal is to detect the dominant scene plane, we find that a fixed value $\epsilon = 4$ (pixels) works well enough in practice. Finally, we employ the standard bundle adjustment to obtain optimal estimates of all homograhpies using the inlying trajectories. Figure 5 shows some representative results of our method.

4.2 Pixel-wise accurate registration via matrix rank minimization

Given the initial estimates of pairwise homographies, we now refine them using our batch alignment formulation (9). For a long video sequence, we can divide the entire sequence into multiple small (overlap) batches of size $(p+1)$, so that the i-th batch contains frames $(p \times (i-1)+1)$ to $(p \times i+1)$, and apply RASL to solve (9) for each batch individually. Note that the way we divide the sequence

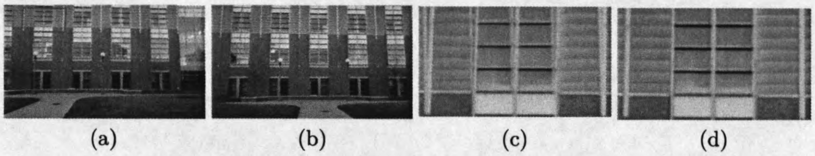


Fig. 6. Pixel-wise accurate registration via matrix rank minimization. **(a) and (b):** The first and last frames of a batch. **(c):** Superposition of the blue window labeled in (a) using frame (a) and (b), according to the homographies estimated by our RANSAC-based plane detection and tracking algorithm. **(d):** Superposition of the same window according to the homographies refined by RASL. The improvement is clear.

ensures that any two adjacent batches share exactly one frame, which enables us to link all the transformations between frames into a common coordinate system in the end[3]. In addition, as suggested by [10], the value of p should be chosen as large as possible, as the low-rank model works better when p is much larger than the dimension of the subspace spanned by the intrinsic views. In our problem, however, p is restricted by the condition that R must be large enough so that (9) can be solved reliably in the presence of gross errors. See Figure 4 for an illustration of the relation between the size of R and the batch size p. In practice, we find that a fixed value $p = 10$ works well for all videos.

In addition, one may notice that our definition of the observable region Ω, and subsequently R in the RASL problem (9), actually depends on τ. To make the RASL algorithm tractable, we pre-compute R using τ estimated by the proposed generalized RANSAC algorithm and fix it in RASL. As a result, some entries in R may become unobservable in certain images as τ changes while running RASL, resulting in some zero entries in the data matrix. However, since RASL is robust to sparse errors, we found that this has little effect, if any, in affecting the accuracy of the alignment results. Figure 6 shows an example of refining the homography matrix using RASL.

5 Experiments

In this section, we report results of our method on both videos captured by ourselves using a hand-held camera (Figure 7) and videos from the Google Map Street View database captured by camera mounted on a moving car (Figure 11). To better understand the advantages of our method, we compare our method against popular image stitching software systems AutoStitch [3] and Photomerge in Adobe Photoshop CS5[4].

Our Video Squences. In Figure 8, we show the results of all three methods on sequence H1. Comparing Figure 8(a) with the corresponding input images, which is the second image of the first row in Figure 7, one can see that objects do

[3] By default, we choose the planar coordinates of the first frame of the sequence as this common coordinate system. All stitching results are represented in this coordinate system, unless otherwise stated.

[4] http://www.adobe.com/products/photoshop.html

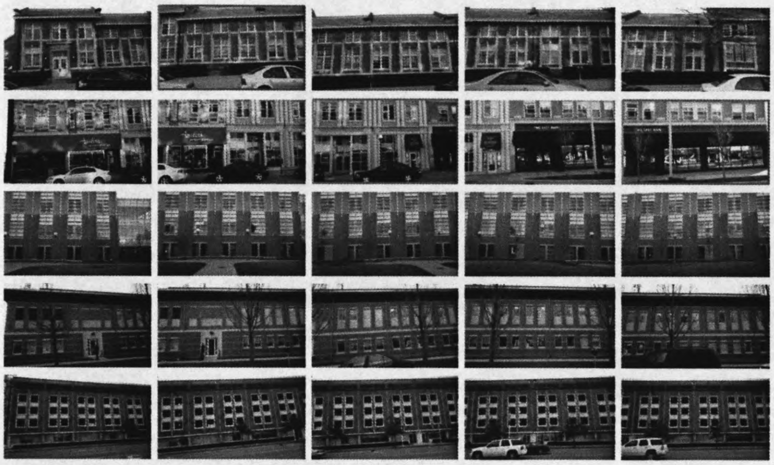
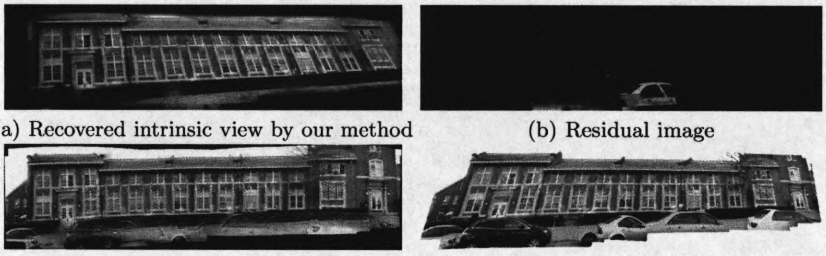


Fig. 7. Snapshots of testing videos taken by a hand-held camera. **From top to bottom:** Sequences H1, H2, H3, H4 and H5. The number of frames for each sequence ranges from 60 to 120.



(a) Recovered intrinsic view by our method

(b) Residual image

(c) Stitching result by AutoStitch

(d) Stitching result by Photomerge

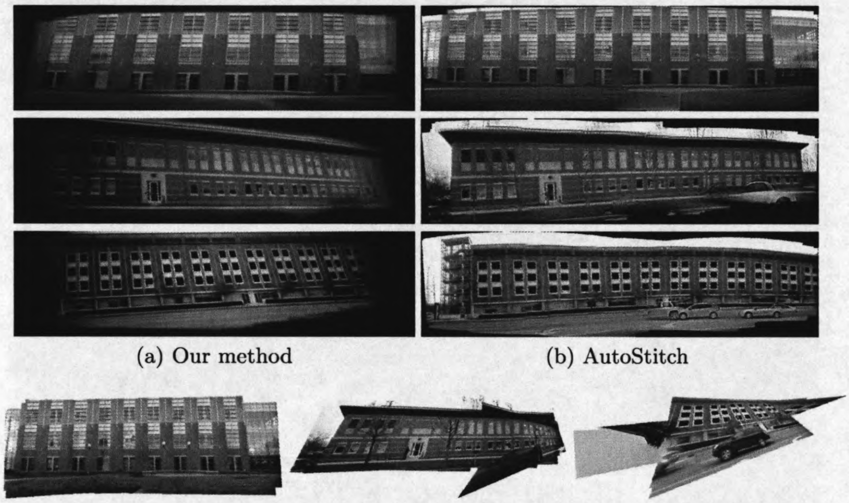
Fig. 8. Video stitching results on sequence H1. The corresponding input image for our method is labeled in blue in Figure 7.

not belong to the dominant plane (i.e. cars) have been completely removed from the intrinsic view by our method. This is also evident by comparing the error map Figure 8(b) with its corresponding frame. However, both AutoStitch and Photomerge fail to handle such outliers properly, resulting in undesired effect of ghosting (Figure 8(c)) or cutting-through (Figure 8(d)). In terms of registration accuracy, it is easy to see that AutoStitch, which relies on detecting and matching SIFT features, performs much worse than both our method and Photomerge in this example.

In Figure 9, we show another example, in addition to Figure 2, where the intrinsic views generated by our method can preserve small depth variations, while occluding objects, such as trees, and reflections being removed simultaneously. The readers are encouraged to see the *supplementary material* for more details about this interesting phenomenon. It is also worth noting that for this sequence

(a) (b) (c)

Fig. 9. Video stitching results on sequence H2. The two corresponding input images are labeled in red in Figure 7. **(a):** Recovered intrinsic views. **(b):** Residual images. **(c):** Details of a small region in (a).



(a) Our method

(b) AutoStitch

(c) Photomerge

Fig. 10. Comparison of video stitching results on sequences H3 to H5.

our method is not able to completely remove cars parking along the street. This is because these cars are very close to the building facades that there is simply not a single frame in which the occluded part of facades becomes visible.

In Figure 10, we show comparative results on sequences H3 to H5. As one can see, our method performs consistently better than AutoStitch and Photomerge, producing clean, pleasing-looking results with pixel-wise registration accuracy. Rather surprisingly, the performance of Photomerge is very unstable, possibly due to its difficulty in matching images in the presence of large repetitive patterns.

Google Map Street View Sequences. Finally, we compare all three methods on the Google Map Street View database (Figure 11). As one can see in Figure 12, our method is very stable on all the sequences, while the other two methods both have obvious problems registering the input images. Furthermore, our method

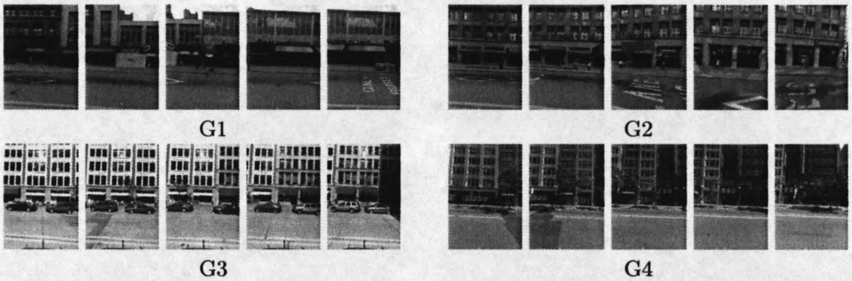


Fig. 11. Snapshots of testing videos from Google Map Street View database. The number of frames for each sequence ranges from 30 to 60.

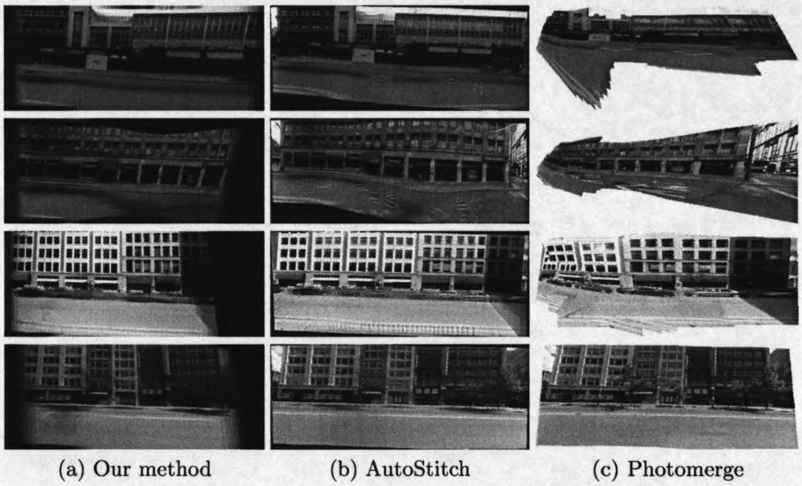


Fig. 12. Video stitching results on Google Map Street View sequences. **From top to bottom:** Sequences G1 to G4.

successfully remove outliers such as reflections on the window (sequences G1, G2 and G3) and trees (sequence G4) from the intrinsic views, while preserving the details on the dominant planes.

6 Discussion and Future Work

In this work, we have shown that by harnessing the inherent relationships across multiple frames of a street view video via a low-dimensional subspace model, the video stitching problem can be solved in a holistic and efficient way, despite exposure differences, parallax, and occluding objects. However, we note that, in fact, much more can be done based on tools we developed in this paper. For example, although our formulation successfully exploits the correlation among multiple views, rich low-rank structures within individual images can further improve the results. Since the street view contains many regular structures, we

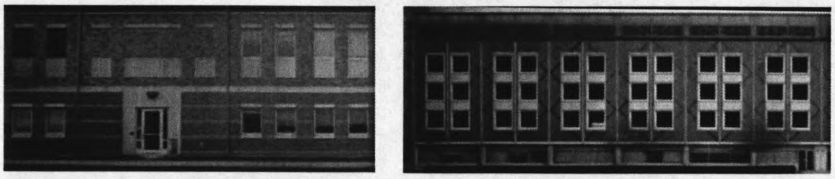


Fig. 13. Rectified panoramas of sequences H4 and H5 using TILT.

could further use tools such as transform invariant low-rank texture (TILT) [18] to rectified the so obtained intrinsic views, as shown in Figure 13. In addition, it is straightforward to extend our method to generate 360° panorama from images taken by a rotating camera, as the frames are related through pure rotations. Finally, the intrinsic views obtained by our method can be used to synthesize many types of new videos (say more stabilized or higher-resolution ones) for the street, which we consider as a promising future direction.

References

1. A. Agarwala, M. Dontcheva, M. Agrawala, S. M. Drucker, A. Colburn, B. Curless, D. Salesin, and M. F. Cohen. Interactive digital photomontage. *ACM Trans. Graph.*, 23(3):294–302, 2004.
2. S. Becker, E. J. Candés, and M. Grant. Templates for convex cone problems with applications to sparse signal recovery. *preprint*, 2010.
3. M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 74(1):59–73, 2007.
4. E. J. Candés, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3), 2011.
5. T.-J. Cham and R. Cipolla. A statistical framework for long-range feature matching in uncalibrated image mosaicing. In *CVPR*, pages 442–447, 1998.
6. A. Eden, M. Uyttendaele, and R. Szeliski. Seamless image stitching of scenes with large motions and exposure differences. In *CVPR*, pages 2498–2505, 2006.
7. M. Irani and P. Anandan. Video indexing based on mosaic representations. *Proceedings of the IEEE*, 86(5):905–921, 1998.
8. Anat Levin, Assaf Zomet, Shmuel Peleg, and Yair Weiss. Seamless image stitching in the gradient domain. In *ECCV*, pages 377–389, 2004.
9. B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, pages 674–679, 1981.
10. Y. Peng, A. Ganesh, J. Wright, and Y. Ma. RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images. In *CVPR*, 2010.
11. P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318, 2003.
12. H. S. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. *TPAMI*, 21(3):235–243, 1999.
13. H. Shum and R. Szeliski. Construction of panoramic image mosaics with global and local alignment. *IJCV*, 36(2):101–130, 2000.
14. D. Steedly, C. Pal, and R. Szeliski. Efficiently registering video into panoramic mosaics. In *ICCV*, pages 1300–1307, 2005.

15. M. Uyttendaele, A. Eden, and R. Szeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *CVPR*, pages 509–516, 2001.
16. J. Wright, A. Ganesh, K. Min, and Y. Ma. Compressive principal component pursuit. *preprint, available at http://arxiv.org/abs/1202.4596*, 2012.
17. W. Xu and J. Mulligan. Performance evaluation of color correction approaches for automatic multi-view image and video stitching. In *CVPR*, pages 263–270, 2010.
18. Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. TILT: Transform invariant low-rank textures. In *ACCV*, pages 314–328, 2010.