

July 2011

UIIU-ENG-11-2205
DC-252

HOLISTIC 3D RECONSTRUCTION OF URBAN STRUCTURES FROM LOW-RANK TEXTURES

Hossein Mobahi, Zihan Zhou, Allen Yang, 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 July 2011		3. REPORT TYPE AND DATES COVERED	
4. TITLE AND SUBTITLE Holistic 3D Reconstruction of Urban Structures from Low-rank Textures				5. FUNDING NUMBERS NSF CCF 09-64215, NAVY N00014-09-1-0230, ARO MURI W911NF-06-1-0076, ARL MAST-CTA W911NF-08-2-0004.	
6. AUTHOR(S) Hossein Mobahi, Zihan Zhou, Allen Yang, 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 ORGANIZATION REPORT NUMBER UILU-ENG-11-2205 DC-252	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) USARO, 2800 Powder Mill Rd. Adelphi MD 20783-1197 ONR, Ballston Centre, Tower 1, 800 N. Quincy, Arlington, VA 22217-5660 NSF, 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) We introduce a new approach to reconstructing accurate camera geometry and 3D models for urban structures in a holistic fashion, i.e., without relying on extraction or matching of traditional local features such as points and edges. Instead, we use semi-global or global features based on transform invariant low-rank textures, which are ubiquitous in urban scenes. Modern high-dimensional optimization techniques enable us to accurately and robustly recover precise and consistent camera calibration and scene geometry from a single or multiple images of the scene. We demonstrate how to construct 3D models of real buildings from sequences of multiple large-baseline uncalibrated images that conventional SFM systems do not apply.					
14. SUBJECT TERMS Holistic, 3D Reconstruction, Low-rank Textures				15. NUMBER OF PAGES 25	
				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 3D Reconstruction of Urban Structures from Low-rank Textures

Hossein Mobahi[†] Zihan Zhou^{*}

hmobahi2@illinois.edu zzhou7@illinois.edu

Allen Yang[◇] Yi Ma^{*}

yang@eecs.berkeley.edu yima@illinois.edu

[†] Department of Computer Science
University of Illinois at Urbana Champaign

^{*} Department of Electrical & Computer Engineering
University of Illinois at Urbana Champaign

[◇] Department of Electrical Engineering & Computer Science
University of California – Berkeley

July 12, 2011

Abstract

We introduce a new approach to reconstructing accurate camera geometry and 3D models for urban structures in a holistic fashion, i.e., without relying on extraction or matching of traditional local features such as points and edges. Instead, we use semi-global or global features based on transform invariant low-rank textures, which are ubiquitous in urban scenes. Modern high-dimensional optimization techniques enable us to accurately and robustly recover precise and consistent camera calibration and scene geometry from a single or multiple images of the scene. We demonstrate how to construct 3D models of real buildings from sequences of multiple large-baseline uncalibrated images that conventional SFM systems do not apply.



Figure 1: **Left Pair:** Example of matched facades of a building. **Right Pair:** Automatically reconstructed 3D model from *only four uncalibrated* images around the building by our method. Each image covers a pair of facades. The pyramids show the estimated location of cameras.

1 Introduction

Recently, there has been tremendous interest in building large-scale 3D models for urban areas, which are largely driven by industrial applications such as Google Earth, Street View, and Microsoft’s Bing Maps, etc. To meet the demands of such applications, **significant progress about the structure-from-motion (SFM) techniques has been made** in terms of the scalability and reliability [1, 22, 17, 7].

The conventional SFM approach to build a 3D model of a scene typically relies on detecting, matching, and triangulating a set of feature points (and edges) in multiple camera views, which has been extensively studied in the past two to three decades. One great advantage of working with point features is that the system can be somewhat oblivious to the scene: the scene could be of any shape or texture as long as the shape is rigid and texture is rich of distinguishable feature points.¹

In practice, researchers have observed that urban scenes often have very special types of shapes and textures, which may not be ideal for generic SFM techniques. On one hand, the shape of man-made objects (e.g., buildings, houses, and cars) normally has very regular global structures, rich of all types of symmetry and self-similarity. If a reconstruction algorithm can take advantage of such global information, it is natural to expect the algorithm to obtain more accurate estimates for both the 3D shape and camera locations from man-made objects than from generic 3D scenes. On the other hand, due to the same reason that urban scenes are full of symmetry, repetitive features pose significant challenges for matching them across different views. The latter problem gets more drastic when the views are sparse and the baseline between views is large as in Figure 1.

In order to handle wide-baseline images for SFM, which represent a large portion of images captured in popular applications [22, 7], more sophisticated techniques have

¹Of course, there have been multiple parallel lines of work in studying 3D shape reconstruction for scenes that lack rich textures, using cues such as shape from shading and contours, etc.

been proposed to extract and match richer image features beyond points and edges. **Examples include affine-invariant features (SIFT)** [13, 15, 16, 2], superpixels [17], and object part-based regions [27, 8], to name just a few. In addition to improving local-feature detection, it has long been noticed that 3D reconstruction of urban structures can be more accurate and simple if one can detect in advance certain salient symmetric patterns (see [12] for a review on this topic) or global structures such as vanishing points [18]. However, symmetry and vanishing points are global or semi-global properties of the scene structures. They cannot be easily extracted from any individual image features. Instead, they must be inferred from the relations among a group of related feature points or edges.

Despite numerous attempts [19, 12], it remains a challenging problem to reliably **detect and extract large, symmetric patterns**. **The reason is twofold: First, it is difficult** to properly detect all the features that represent a symmetric pattern (say the four corners and four edges of a window). **Second, the task becomes more daunting** in the presence of outliers and partial occlusion in the extracted feature set, which obscure the dominant global structures. This is the main reason robust statistical techniques such as Hough transform or RANSAC have been widely used for such purposes. Finally, even when local features are reliably extracted, it is not trivial to verify which ones satisfy what symmetric and/or vanishing point relations under camera perspective projection [10]. To address these problems, there has been increasing amount of work trying to infer approximate 3D geometry of image patches of urban scenes using supervised machine learning methods [11, 9, 21]. In contrast, in this paper, we investigate a novel approach to infer accurate 3D geometry from either a single or multiple images of an urban scene in a totally unsupervised fashion.

We believe part of the reason why inferring 3D geometry from local features has become a bottleneck is that all the rich global geometric relations have been largely ignored when they are extracted *locally* from the images. Most existing algorithms only try to reassemble such global relations back after the features have been extracted *individually*. **Therefore, if we want to avoid aforementioned difficulties, we must find new “building blocks”** for modeling urban objects that complement local features such as points, edges, SIFT features, and generic local patches. The new building blocks shall have the following good properties:

1. *Holistic*: They need to encode accurate, global geometric information such as structural symmetry, vanishing points, and camera positions;
2. *Invariant*: Their representation should be invariant to camera viewpoint and perspective distortion, so that they can be matched reliably across multiple images;
3. *Robust*: The extraction of such new features should be numerically stable and robust (say, to partial occlusion or random image noise and error).

Contributions. Recently, a new type of image feature called transform invariant low-rank texture (TILT) has been proposed by [26]. **We find that it meets the above requirements.** The goal of this paper is to demonstrate how such low-rank textures can be used as new building blocks for modeling urban scenes. The proposed new approach suggests that we can obtain accurate 3D models for urban objects such as buildings and



Figure 2: Geometry from a low-rank patch on a building facade. **Left:** The red box represents the selected candidate region I , and the green box corresponds to the recovered low-rank texture using TILT. **Right:** The rectified building facade $I_0 = I \circ \tau$ using the homography τ estimated from the low-rank texture.

houses, *without relying on extraction of any traditional local features such as points and edges*. The new approach relies directly and exclusively on semi-global or global image patches and regions built from TILT features. For this very reason, the approach is called “holistic.” We show how to obtain accurate information about camera calibration, orientation and position from each image, correspondence between two images, and ultimately a consistent 3D structure from multiple images, as the example shown in Figure 1.

Admittedly, the proposed basic scheme cannot yet handle all comprehensive urban scenes, especially where low-rank texture is not abundant. Therefore, it should not be treated as a replacement or competitor to the existing SFM systems. Rather, the interesting new tools introduced in this paper are more tailored to regular urban objects, and they should be considered complementary to existing general-purpose point-feature based SFM methods.

2 Geometry from One Facade of a Building

For completeness, we first give a brief review of work on low-rank textures [26] and then show how to use them for 3D modeling. It has been observed by the authors of [26] that the image of repetitive or symmetric patterns, when viewed as a matrix, is *low-rank*. For example, if I_0 is a rectified frontal view of a planar patch on the facade of a typical office building (see Figure 2 right), then as a matrix, I_0 has a rank much lower than its dimension. The authors call such an image patch as a *low-rank texture*. In some other (perspective) view of the building (see Figure 2 left) the corresponding patch I deforms by a homography transform: $I = I_0 \circ \tau^{-1}$, where τ belongs to the homography group $GL(3)$ and deforms the image domain.

One intriguing observation of the work [26] is that as long as the patch is large enough and contains sufficient texture, both the deformation τ and the view-invariant low-rank texture I_0 can be accurately recovered from the observed I , up to scaling in each of the image coordinates. The basic idea is to solve for a transformation τ of I

so that $I_0 = I \circ \tau$ has the lowest possible rank. Furthermore, the image patch I is often corrupted by noise and occlusion. As a result, a more realistic model between the low-rank texture I_0 and its image I has been proposed by [26] as:

$$I \circ \tau = I_0 + E, \quad (1)$$

where E represents some sparse error that corrupts the image, say due to partial occlusion. As shown in the work [26] and Robust PCA literature [5], the transformation τ and the low-rank texture I_0 can be recovered by solving the following optimization problem:

$$\min_{A, E, \tau} \|A\|_* + \lambda \|E\|_1 \quad \text{subject to} \quad I \circ \tau = A + E, \quad (2)$$

where $\|\cdot\|_*$ and $\|\cdot\|_1$ represent the nuclear norm and ℓ_1 -norm of a matrix, respectively².

The recovered low-rank texture A only differs from the original low-rank texture I_0 by a scaling in the x and y coordinates. The recovered τ encodes the homography from the default image plane $z = 0$ to the low-rank planar patch in 3D: $\tau^{-1} \doteq [t_1, t_2, t_3] = K[n_1, n_2, T]$, where $R = [n_1, n_2, n_3] \in \mathbb{R}^{3 \times 3}$ is the rotation, $T \in \mathbb{R}^3$ the translation, and $K \in \mathbb{R}^{3 \times 3}$ the intrinsic parameters of the camera. If the horizontal and vertical directions of the low-rank patch are parallel to two vanishing directions in 3D, then **the first and second columns of τ^{-1}** as a 3×3 matrix give the coordinates of the two vanishing points $v_1 = t_1, v_2 = t_2 \in \mathbb{R}^3$ in the image coordinates, respectively [14]. If the camera is calibrated, the two vanishing points should be orthogonal to each other. So from a low-rank texture region in an uncalibrated image, we obtain one linear constraint on the camera intrinsic parameters $K \in \mathbb{R}^{3 \times 3}$: $v_1^T K^{-T} K^{-1} v_2 = 0$. If the **image(s) consist of a sufficient number (≥ 5)** of low-rank patches with independent orientations in 3D, one can fully recover the camera intrinsic parameters K without any special calibration apparatus.

3 Geometry from Intersecting Facades

Although the TILT method allows us to extract geometry from each individual low-rank patch, an urban scene typically consists of numerous low-rank regions. A representative image of a building may contain two or more of its facades, as shown in Figure 3(a). The homographies recovered from individual patches on each of the facades may not be consistent in their scales.

Normally the low-rank textures on two intersecting facades of a building give three sets of parallel lines, two horizontal and one vertical. These three sets of parallel lines **define three vanishing points in the image, denoted as $v_1, v_2, v_3 \in \mathbb{R}^3$** , respectively. Notice that the pairs (v_1, v_3) and (v_2, v_3) can be obtained from the homography of an individual low-rank patch on each of the facades.

In order to determine the relative scale of the two facades in 3D, we need to find their intersection line l in the image. It belongs to the one-parameter family of lines

²The nuclear norm of A is defined as the sum of its singular values: $\|A\|_* = \sum_i \sigma_i$. The ℓ_1 -norm of E is defined as $\|E\|_1 = \sum_{i,j} |e_{ij}|$.

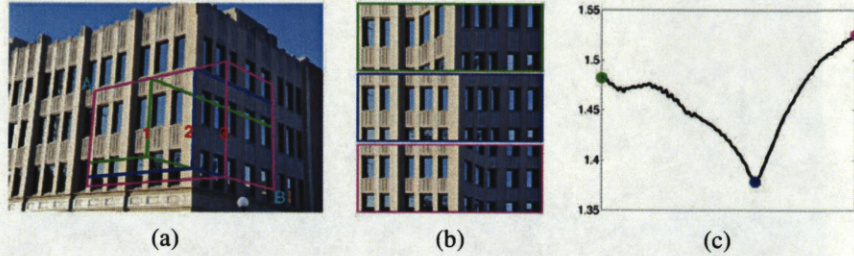


Figure 3: Identifying the intersection line l of two facades. (a) Three different intersection hypotheses for the two adjacent 4-sided polygons. (b) The unfolded joint textures: each corresponds to one of the hypotheses shown in (b), as indicated by the color of its boundary. (c) The value of (3) as a function of the location of the intersection line. It is minimized precisely at the correct location (the blue line).

that go through the vanishing point v_3 in the image. As it turns out, we can use the *joint low-rank structure* of both facades to determine the precise location of this line regardless whether there is a visible edge along this line or not.

To see this, let us fix one point in each facade, say, the upper-left corner of a low-rank patch on the first facade and the lower-right corner of a patch on the second facade, labeled as points A and B respectively, as shown in Figure 3(a). As one can see, since the vanishing points v_1, v_2, v_3 are known, any intersection line l between A and B will uniquely determine a special structure with two adjacent 4-sided polygons in the image, each corresponds to a facade of the building. That is, the homographies τ_1 and τ_2 of the two facades are parametrized by the same one-parameter family lines l passing through v_3 . Figure 3(a) shows examples of the special structure with three different intersection lines (labeled as 1, 2 and 3).

Given the corresponding homography $\tau_i(l)$, we may rectify each polygon and then concatenate them into a joint rectangular texture, as shown in the Figure 3(b). Then the joint texture, as a matrix, should also have the lowest rank when the intersection line is the correct one (Figure 3(c)).

Mathematically, let I_1 and I_2 be the two low-rank texture windows subject to transformations τ_1 and τ_2 , which depend only on l . **We find the true position of the intersection line l^* by solving the following optimization problem:**

$$\begin{aligned}
 l^* &= \arg \min_l \|[A_1 \ A_2]\|_* + \lambda \|[E_1 \ E_2]\|_1 \\
 \text{s.t. } &I_i \circ \tau_i(l) = A_i + E_i, \quad i = 1, 2.
 \end{aligned} \tag{3}$$

This problem can be solved very effectively using a line search technique along the unknown parameter l . Figure 3(c) shows a typical plot of values of the objective function. Once the intersection line l^* is determined, the relative scale of the two facades and camera geometry are uniquely determined. Figure 4 shows more representative results. **As one can see, the proposed scheme accurately identifies the correct intersection lines even when local edge features around the intersection, on which most traditional methods rely, are almost *invisible* in the image (e.g., in Figure 4(b)) or even suggest an**

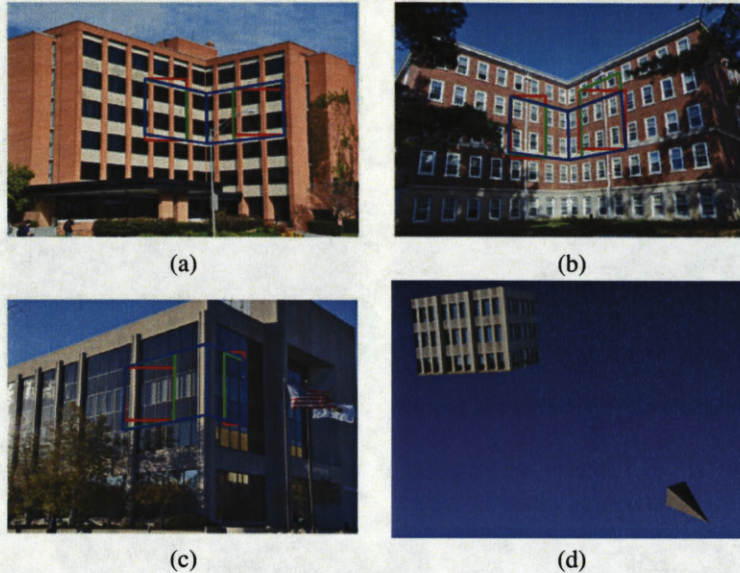


Figure 4: **(a)–(c)**: Additional representative results of identifying the intersection line of two adjacent facades. Red windows are the initialization. **(d)**: Accurate 3D “pop-up” from the single image in Figure 3. Camera position is recovered, shown as a small pyramid.

incorrect line (e.g., in Figure 4(c))!

If the camera is calibrated, from the assumption we know v_3 should be orthogonal to both v_1 and v_2 as $v_3 \sim v_1 \times v_2$. Very often, the two facades of the building are also orthogonal to each other, i.e., $v_1 \perp v_2$.³ If the camera is not calibrated, the three vanishing points impose three independent constraints on the camera intrinsic parameters:

$$v_1^T K^{-T} K^{-1} v_2 = 0, v_1^T K^{-T} K^{-1} v_3 = 0, v_2^T K^{-T} K^{-1} v_3 = 0.$$

This allows us to fully calibrate the camera from just a pair of intersecting facade, if only the focus length f and principal point (o_x, o_y) are not known in K . An example of such reconstruction from single image is shown in Figure 4 (d).

4 Segmenting Building Facades

Patches of low-rank textures allow us to extract from a single image accurate information about the camera location, calibration, and 2D textures and 3D structures. But in order to obtain a complete 3D model from multiple images around a large building, we need to establish correct, precise point-wise correspondence between different views.

³This may not always be the case. For instance, the facades in Figure 10 (a) and (b) are not orthogonal.

Repetitive features and patterns in an urban scene make finding the correct correspondence between images much more challenging than that for a generic non-urban scene. The reason is obvious: Matching local features or even local patches are inherently ambiguous – there are many other points and patches in the other image(s) that have exactly the same local appearance. Most SFM methods then rely on having images taken with relatively small baselines, either from a video sequence or from a very dense set of photos.

When the baseline between images is large or images are sparse, any effort to eliminate such ambiguity has to rely on certain global spatial relationships among multiple points, lines, or patches. The approach we propose here relies on a very simple observation: *the larger the patch or region we match, the less the ambiguity* [25, 24]. To the extreme, if we can detect the entire facades, then the matching would have minimal ambiguity. Hence, a necessary step to establish globally consistent correspondence between views is to segment out each building facade.

As different facades of the same building often have the same local color and textural appearance (see Figure 4), global geometry and texture become the only cues to tell them apart. Our approach relies on another simple observation: *if two adjacent patches, say I_1, I_2 , belong to the same facade, then after we merge them into a larger patch $I = [I_1, I_2]$, the joint texture should remain low-rank (after rectification by a homography found by TILT: $I \circ \tau = A + E$).* Such a patch I can be represented very compactly by the triplet (A, E, τ) : the homography τ , the low-rank component A , and the sparse component E . Thus, by comparing the compactness of the representation before and after the merging, we can tell whether the two patches belong to the same facade or not.

In the rest of the section, we first derive a purely objective measure for the compactness of a patch I based on its coding length⁴, and then we show how to use this measure to effectively cluster patches to form facades.

4.1 Compact Coding for Low-rank Textures

A naive way to encode the patch I would be entropy-coding of the quantized sequence of pixel values in I , as conventional image compression schemes do. However, when $\text{rank}(A)$ is small and E sparse, encoding I in terms of the triplet (A, E, τ) is far more efficient as both sparse and low-rank matrices allow efficient coding. **In order to get a finite coding length, the components of the triplet must be quantized.** Denote the number of bits required to represent a quantized real number by f .⁵ For controlling overall reconstruction quality of the patch, we define a distortion parameter ϵ . No matter how we encode the patch, the decoded triplet $(\hat{A}, \hat{E}, \hat{\tau})$ must satisfy a distortion

⁴There is a theoretical connection between rank and the coding length of a matrix [20]. However, rank is very sensitive to noise and outliers. We have conducted experiments using the aggregated rank, and the segmentation results are unstable. The proposed coding length is essentially a robust measure of rank based on Robust PCA of the image region.

⁵We have empirically observed that for any real number, 16 bits are more than sufficient to ensure a good precision. For example, the homography τ is a 3×3 matrix. Thus, it is sufficient for us to assign $9f$ bits to it, i.e. $L(\hat{\tau}) = 9f$, where $\hat{\tau}$ is the quantized τ .

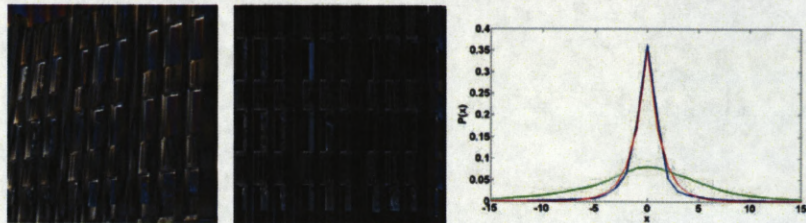


Figure 5: **Left:** The residual matrix \hat{E} of the original (left) and transformed (right) images in Figure 6 approximated by top three singular values/vectors. **Right:** Empirical probability distributions of the errors for the left (green) and right (blue) residual maps. The empirical distribution (blue) of the right residual map can be fit closely by a Laplace distribution (red).

tolerance:

$$\|(\hat{A} + \hat{E}) \circ \hat{\tau}^{-1} - I\|_F^2 \leq \epsilon^2 \text{size}(I), \quad (4)$$

where $\text{size}(I)$ is the number of pixels of I , say $m \times n$.

Encoding the Sparse Matrix E . The sparsity in \hat{E} implies that it has a very low-entropy – many entries are zero. It has long been observed empirically in signal processing that most sparse signals obey a *Laplace distribution* [4]: $p(x) = \frac{1}{2\lambda} \exp(-\frac{|x-\mu|}{\lambda})$, where we typically assume $\mu = 0$ in our setting. Since we here are working with a set of discrete samples: $\mathcal{X} = \{x_1, \dots, x_N\}$ ⁶, we can work with a discrete Laplace distribution $p_k = \frac{1}{2\lambda} \exp(-\frac{|x_k|}{\lambda})$, over some support interval $[-B, B]$. Here Z is the normalization constant and x_k is a sampling point. We simply choose $B = \max_i |x_i|$ over the sample set \mathcal{X} . The maximum likelihood estimate of λ based on \mathcal{X} is given by the following expression: $\lambda = \frac{1}{N} \sum_{i=1}^N |x_i|$.

Figure (5) shows a typical example of empirical distribution of \hat{E} (blue), from one of the building images, against the estimated distribution $\{p_k\}$ (red). The distribution $\{p_k\}$ has two parameters, namely (B, λ) . Thus, by merely transmitting B and λ , which takes $2f$ bits, the receiver can construct $\{p_k\}$ and use it to infer the optimal codebook for \mathcal{X} . With such a (Laplace) encoder, the expected coding length for \hat{E} would be:

$$L(\hat{E}) = 2f + mn \left(\sum_k -p_k \log_2 p_k \right). \quad (5)$$

Encoding the Low-rank Matrix A . Naive entry-wise encoding of the $m \times n$ quantized low-rank matrix \hat{A} would take mnf bits. However, since A is low-rank, the singular value decomposition $A = U\Sigma V^T$ leads to a more effective encoding. Let $r = \text{rank}(A)$. Then, we only need to encode $(m+n+1)rf$ bits associated with (quantized) non-zero singular values and their corresponding singular vectors: $\hat{A} = \sum_{i=1}^r \hat{\sigma}_i \hat{u}_i \hat{v}_i^T$, where the non-quantized variables are $u_i \in \mathbb{R}^m$, $v_i \in \mathbb{R}^n$ and $\sigma_i \in \mathbb{R}_+$. Obviously, for $r \ll \min\{m, n\}$, this encoding uses much fewer bits than the naive encoding $(m+n+1)rf \ll mnf$.

⁶Each $x_i \in \mathcal{X}$ is an element of the matrix \hat{E} , thus $|\mathcal{X}| = N = mn$.

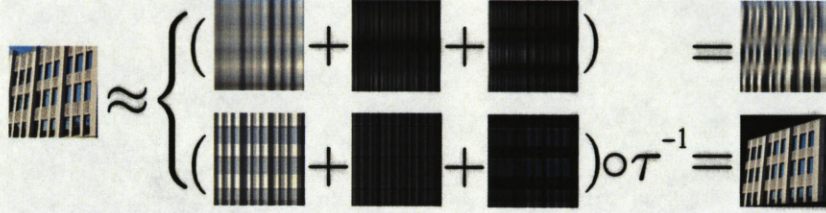


Figure 6: Approximation of a facade image with the top three singular components $\sum_{i=1}^3 \sigma_i \mathbf{u}_i \mathbf{v}_i^T$. **Top:** SVD of the original image. **Bottom:** SVD of the rectified image by TILT.

For noisy real images, A may not be a perfectly low-rank matrix. So we only need to encode its leading rank- q approximation: $A_q = \sum_{i=1}^q \sigma_i \mathbf{u}_i \mathbf{v}_i^T$ subject to the allowed distortion ϵ . The coding length of \hat{A}_q is thus given by:

$$L(\hat{A}_q) = (m + n + 1)qf. \quad (6)$$

We can further compress the vectors $\{\hat{\mathbf{u}}_i, \hat{\mathbf{v}}_i\}$ based on additional structures in them. As each $\hat{\mathbf{u}}_i$ or $\hat{\mathbf{v}}_i$ is often a smooth signal except at the image edges, (see Figure 6), we can encode each vector by a *difference code*⁷ plus the head element. This way, the difference of each vector is a sparse sequence and can be encoded again by the Laplace code.

4.2 Compression-Based Facade Segmentation

To summarize, the coding length required to encode a supposedly low-rank patch I subject to the distortion tolerance ϵ is given by:

$$\min_q L(\hat{A}_q) + L(\hat{E}) + L(\hat{\tau}) \text{ s.t. } \|(\hat{A}_q + \hat{E}) \circ \hat{\tau}^{-1} - I\|_F^2 \leq mn\epsilon^2,$$

where $(\hat{A}, \hat{E}, \hat{\tau})$ is the decoded quantized version of (A, E, τ) .

For an image that contains multiple facades, we segment the image I into a set of subregions $\mathcal{S} = \{I_k\}$ whose union covers all the valid TILT features in I . The goal is to choose \mathcal{S} such that, when each I_k is encoded by the proposed scheme, the total coding length becomes minimal:

$$\begin{aligned} \min_{\mathcal{S}, \{q_k\}} \sum_k L(\hat{A}_{k, q_k}) + L(\hat{E}_k) + L(\hat{\tau}_k) \\ \text{s.t. } \|(\hat{A}_{k, q_k} + \hat{E}_k) \circ \hat{\tau}_k^{-1} - I_k\|_F^2 \leq mn\epsilon^2, \forall k. \end{aligned} \quad (7)$$

We solve this problem in a greedy and agglomerative fashion, similar to that in [20]. The algorithm starts from a simple grid on I , where each I_k is a tile of the grid. At each subsequent iteration, a pair of adjacent regions are chosen to be merged into a larger region, which leads to maximal reduction in the total coding length (7). The process stops when the number of bits can no longer be reduced given the distortion. Figure 7 shows some representative results.

⁷The code subtracts from each element in the sequence the value of the previous element.

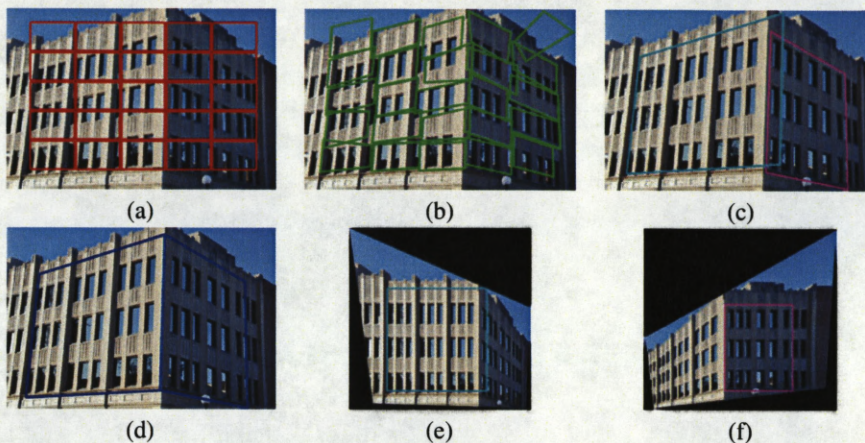


Figure 7: (a): Initial grid. (b): Initial TILT. (c): Final segmented regions. (d): Recovered intersection line. (e)-(f): The homography estimated from cyan and magenta regions applied to the entire I to get the transformed images I' (corresponding regions are rectified).

Comparison with Symmetry Detection. Conceptually, one could also utilize the work of [19] to parse the building facades, which can effectively detect and segment regions tiled by a repetitive 2D pattern. We have tested their method on our data and found that the method is not suitable for our purposes due to several reasons: it often breaks one facade into multiple disconnected small lattices; the symmetry groups/lattices detected from different images (of the same facade) can be very different, and it cannot handle large perspective distortion. We have shown some results in the appendix for comparison. These problems make the results hard to use for subsequent matching.

5 Point-wise Matching of Building Facades

The segmentation provides a good estimate for the relative location of the facades and their rectified texture (see Figure 7 (e) and (f)). Using such rectified textural regions, solving wide-baseline correspondence between two images I_1 and I_2 becomes better conditioned (say by a similarity match). However, each segmented region may not share the same location and scale in different images. Therefore, we need to refine their location and scale in order to obtain precise point-wise matching between images.

Denote A_1 as a low-rank texture from one facade in the first image I_1 . If we assume the triplet (A_2, E_2, τ_2) in the second image I_2 best matches A_1 among all obtained segments in I_2 , then the entire image I_2 can be rectified by the homography τ_2 , and the sparse error E_2 be removed before matching. Thus, the problem is reduced to matching A_1 to a cleaned image: $I'_2 = I_2 \circ \tau_2 - E_2$ (see Figure 7).

The goal now is to find a region R^* in I'_2 which, after translation and scaling, best matches A_1 point-wise. We use normalized cross correlation (NCC) to measure the

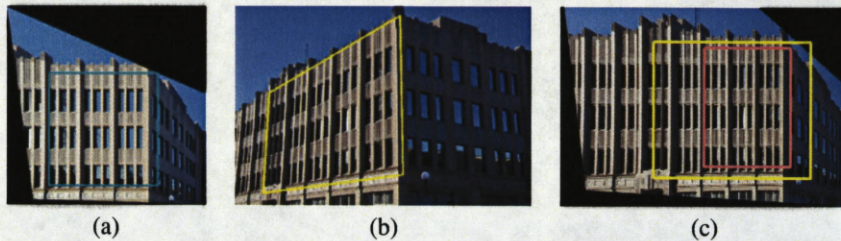


Figure 8: (a) Segmented and unwarped facade. (b),(c), Segmented and unwarped region of the same facade in a different image. In (c), the segmentation result is further refined to the orange box by matching.

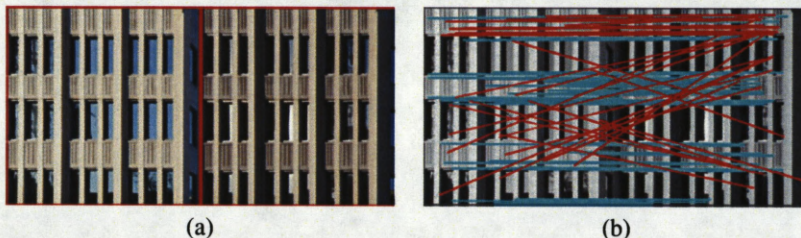


Figure 9: (a) Point-wise match between two regions of the facades. (b) Feature-point matching result of the two rectified regions by SIFT [13], with red lines indicate mismatches.

similarity between the two regions, which is ideal for our task as the regions are already distortion-free. Therefore, the best region is given by the following optimization:

$$R^* = \arg \max_{\phi=(x,y,u,v)} \frac{\text{vec}(A_1)^T \text{vec}(R \circ \phi)}{\|\text{vec}(A_1)\|_2 \|\text{vec}(R \circ \phi)\|_2}, \quad (8)$$

where ϕ is parameterized by the center location (x, y) of R and scales (u, v) in x and y directions, respectively.

We solve the optimization task iteratively. Initially, we start from a guess (x_0, y_0, u_0, v_0) , which is a box among the candidate regions in I_2 (such as those in Figure 7) that has the highest NCC with A_1 . We then maximize the objective function in a gradient ascent fashion. The iteration terminates when no more improvement can be made. Due to the greedy nature of this procedure, theoretically we can only guarantee a local optimal matching region \hat{R} . However, since we are working with very large segmented regions, we have observed in practice that the above procedure typically finds the globally optimal matching. Again, since there is no geometric distortion left in the rectified low-rank textures, the refinement converges to a very precise point-wise matching. If the two images each has multiple (segmented) facades, we run the above matching procedure on each candidate pair and choose the one that has the best matching score. As the number of segments is typically very small (2 or 3 per image in most cases), this process is very efficient.

Comparison with Feature Matching. An example of final matching results between two images are given in Figure 8. As a comparison, in Figure 9, we illustrate the difficulty of applying the classical SIFT matching technique [13] to the same urban scenes with repetitive or symmetric patterns. Point-wise matching of low-rank regions outperforms SIFT in this scenario because the texture segmentation results enable us to perform accurate region-based matching rather than using local points or edges.

6 Full 3D Reconstruction of Buildings

In this section, we demonstrate how the techniques from the earlier sections can be assembled together for 3D reconstruction of a large octagonal building.⁸ We use only eight *uncalibrated and widely separated* images for the full reconstruction of the building. Each of the images covers a pair of adjacent facades as shown in Figure 10. This building has a few interesting properties. First, the large number of facades and intersections will magnify the accumulation of (geometry or calibration) error if any. **Second, occlusion by trees and reflections on the glass are two major problems that challenge conventional SFM methods, but can testify the robustness of our scheme against such errors.**

We do not use any prior information about the geometric model of the building except that all the facades share the same vertical vanishing point. We use the vanishing point constraints to partially determine the calibration matrices of the eight images. Since two facades in each image impose two independent constraints on the calibration matrix, we use them to recover the focal length f and the x -coordinate o_x of the principal point, assuming the y -coordinate o_y is fixed at one half of the image height. Once the calibration matrix is obtained, we can compute the relative orientation and position of the camera with respect to the scene.

To segment the facades, we assume the rough location of the building within the images is provided.⁹ A 5×5 grid of initial windows is then placed around this location. **Some of the identified facades for the octagonal building are shown in Figure 10(a) and (b).** We further arrange the sequence of images so that matching of common facades is only performed between consecutive images. See Figure 10(c) and (d) for an example of the matched facades and appendix for additional details.

Now we can obtain a full 3D reconstruction by assembling the building one view at a time using consecutively matched facades. However, errors in both camera parameters and the 3D model, when estimated from real images, are inevitable. For example, the camera calibration may not be precise enough because of simplifying assumptions on the parameters (i.e., f, o_x, o_y). Thus, if we assemble the views one by one, geometric error accumulates as the number of images increases. For example, the start and the end of the model do not meet each other in Figure 10(e).

Enforcing Global Consistency. For global consistency, we propose a global objective, which uses the current camera parameters and 3D model as the input, and tries

⁸For 3D reconstruction of a typical rectangular building, see Figure 1.

⁹Either by the user or by a simple detection scheme.

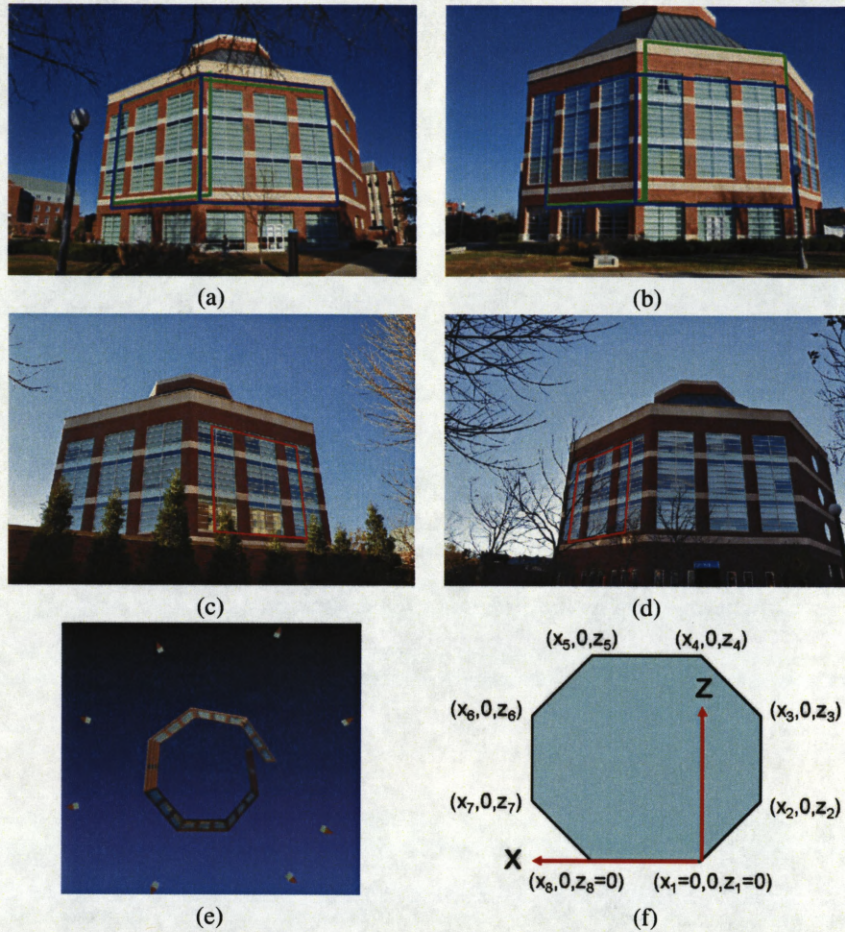


Figure 10: (a) and (b): Segmentation (green) and intersection detection (blue) on two images of an octagonal building. (c) and (d): A pair of matched regions from the same facade with different partial occlusion. (e): A top view of the reconstructed structures of the octagonal building showing the accumulated geometry error when assembling the views one by one. (f): The parameterized 3D model of the building.

to refine them simultaneously. Conceptually, this is similar to “bundle adjustment” in conventional SFM.

We randomly select two adjacent facades, say the pair in Figure 10(a), and choose the origin of the world frame to be a point at the intersection of the two facades. In addition, we let the x and y axes of the world frame to be parallel to the left facade in that image. Once the world frame is chosen, a building with n facades can be described using a set of n points $X = \{X_i\}_{i=1}^n$, where each $X_i = (x_i, 0, z_i)^T$ is (1) on the plane $y = 0$ and (2) at the intersection line of two adjacent facades. These points form a



Figure 11: Top and frontal views of the recovered building. Each pyramid shows the estimated location of a camera.

n -sided polygon on the $y = 0$ plane. For example, the 3D model of the octagonal building ($n = 8$) is shown in Figure 10(f).

For the cameras, we use the same set of parameters $\{K_i, R_i, T_i\}_{i=1}^n$ as before. Here we assume both the focal length f_i and the principal point (o_{x_i}, o_{y_i}) of each

camera are unknown. Now we formulate the global optimization as follows. First, from each image I_i , we can extract two rectified facades (A_i^j, E_i^j) , $1 \leq j \leq 2$: $I_i \circ \tau_i^j(K_i, R_i, T_i, X) = A_i^j + E_i^j$. Second, we ask the i -th pair of matching facades to be the same, up to some sparse error e_i :

$$I_i \circ \tau_i^2(K_i, R_i, T_i, X) = I_{i'} \circ \tau_{i'}^1(K_{i'}, R_{i'}, T_{i'}, X) + e_i, \quad (9)$$

where $i' = \text{mod}(i + 1, n)$. Combining these two criteria, we propose to solve the following problem:

$$\begin{aligned} \min \sum_{i=1}^n \sum_{j=1}^2 \{ \|A_i^j\|_* + \lambda \|E_i^j\|_1 \} + \sum_{i=1}^n \gamma \|e_i\|_1, \\ \text{s.t. } I_i \circ \tau_i^j(K_i, R_i, T_i, X) = A_i^j + E_i^j, \\ I_i \circ \tau_i^2(K_i, R_i, T_i, X) = I_{i'} \circ \tau_{i'}^1(K_{i'}, R_{i'}, T_{i'}, X) + e_i, \end{aligned} \quad (10)$$

where λ and γ are the weights of the respective term. To deal with the nonlinear constraints in (10), we use an iterative scheme, which repeatedly solves the linearized version of (10) w.r.t the current estimates of all unknown parameters $(K_i, R_i, T_i, X)_{i=1}^n$. To reduce the effect of change in illumination and contrast, we normalize each $I_i \circ \tau_i^j$ to zero mean and unit Frobenius norm. With the initialization obtained from assembling the views one by one, the iterative scheme usually converges in 15 to 20 iterations.

Figure 11 shows the reconstructed full 3D model as well as the recovered camera poses. The readers should note the improvement in the top view of the 3D model, compared to Figure 10(e). We also calculated the average error in the eight angles between the building facades. It is 3.1 degree and 1.5 degree before and after global adjustment, respectively. As one can see, despite unknown calibration, partial occlusion, large baselines, our method is able to recover a very precise and complete 3D model of the building.

Comparison with other SFM Systems. It is difficult to make a fair comparison between the proposed approach and other SFM methods, since the large baselines and rich symmetry makes other methods fail. In fact, we tested our sequences on almost all publicly available SFM packages such as Bundler [22], SFM-SIFT¹⁰ (which combines Torr's SFM toolbox [23] with SIFT feature detector [13]), FIT3D [6], and Voodoo Camera Tracker¹¹. All these packages report errors related to their inability of establishing meaningful correspondence across the views. More details about these failures are discussed in the appendix.

7 Conclusion

In this paper, we have illustrated a novel approach to perform holistic 3D reconstruction of urban scenes, directly from low-rank textural regions without extracting any low-level local features. In man-made environments where low-rank and symmetric

¹⁰http://homepages.inf.ed.ac.uk/s0346435/projects/sfm/sfm_sift.html

¹¹<http://www.digilab.uni-hannover.de/docs/manual.html>

textures are abundant, rich geometric information about the camera location and scene structures are accurately encoded in the images. This dramatically simplifies many problems previously deemed difficult: 3D reconstruction from a single view, matching repetitive structures across large baseline. This paper introduces some new computational tools for obtaining such information accurately, efficiently, and robustly, which we believe, highly complement conventional point-based SFM techniques.

8 Acknowledgement

This research was supported in part by NSF CCF 09-64215 and NAVY N00014-09-1-0230, ARO MURI W911NF-06-1-0076, and ARL MAST-CTA W911NF-08-2-0004. Hossein Mobahi was supported by Computational Science & Engineering (CSE) Ph.D fellowship of University of Illinois at Urbana Champaign. The views and conclusions contained in this document are those of the authors and should not be interpreted as **representing the official policies, either expressed or implied, of the Army Research Laboratory or U.S. Government.** The U.S. Government is authorized to reproduce and distribute for Government purposes notwithstanding any copyright notation hereon.

9 Appendix

This appendix contains additional comparison examples of our method with others.

9.1 Segmentation Comparison

We compare our segmentation component against lattice detection work [19]. The latter tries to detect and segment regions tiled by a repetitive 2D pattern in the 3D space. This is relevant as most building facades are comprised of such patterns. The **method first generates multiple lattice hypotheses by clustering some interest points** and then grows the hypotheses in an iterative fashion. When applying to our data, however, we observe a couple of shortcomings. First, the facades are only “partially” detected, which makes the local matching fail due to the ambiguity. In addition, the detected lattices for the same facade in different images usually have different shape and size, which further complicates the matching. Finally, some of the facades with large perspective distortion are not detected at all. See Figure 12 for some examples.

9.2 Comparison with other SFM Systems

As mentioned in the paper, it is difficult to make a fair comparison between our method and some of the existing SFM systems, mainly due to the large baseline and repetitive structures in our data. These properties cause any existing package that we tried to fail, due to their inability to establish meaningful correspondence. In this section, we present a more detailed description about the failures of these systems.

See Figure 13 for an example of SIFT matches obtained by SFM-SIFT [3] (which simply applies Lowe’s original SIFT algorithm [13] followed by Torr’s SFM code). Since the initial correspondence is very poor, the subsequent matching results using SFM are erroneous. There were other indicators of failure as well. For example, focal length computed by [3] becomes a complex number with real component equal to zero. However, the aforementioned problems do not occur on small baseline images with this package.

Voodoo Camera Tracker¹² is another SFM package we have tried. This package works reasonably well on the sample images provided by the package. However, it crashes when our image data are applied (see Figure 14), even when we resized our images to the size of the images in the example sequence of the package.

Bundler [22] also takes SIFT feature matches as the input. It typically requires a large number of matches in order to robustly estimate an initial two-frame reconstruction (at least 100 matches as suggested in [22]). As one can see in Figure 13, the number of SIFT feature matches for adjacent frames in our sequence is very small (usually less than 20), hence results in early stopping of the algorithm without recovering any 3D model or camera parameters. Similar problem arises in FIT3D package [6]

¹²<http://www.digilab.uni-hannover.de/docs/manual.html>



Figure 12: Comparison the lattice detection [19] (left) against our compression based segmentation (right). In the right images, green shows the segmentation result and blue is the refined solution after intersection detection.

9.3 Additional Results on Large-Baseline Matching

In Figure 15 we present some matching results obtained by our method. In addition to the wide baseline, some facades are partially occluded by tree branches. In addition, reflection on window glasses in some images makes matching more challenging.

9.4 Conclusion

In this manuscript, we have compared the performance of the proposed algorithms in segmentation and large-baseline matching against several well-established solutions. We have observed that when the existing methods are applied to the image data that contain large-baseline transformation *and* moderate image corruption, they fail to produce good segmentation, matching, and reconstruction results. These observations in fact are not inconsistent with the past results in the literature as they mostly work well

on small-baseline images with minimal image corruption. The extensive experiment **justifies the novelty and efficacy of the proposed algorithms, when low-rank texture** regions are present in the images and they convey accurate information of the 3-D rigid structures. Finally, we believe that the new techniques complement those more conventional point-based SFM techniques. More sophisticated solutions that would carefully combine the advantages of both approaches should achieve better SFM results in the future.

The work also contains a **video** which illustrates the optimization step for point-wise matching (section 5 of the paper).

References

- [1] A. Akbarzadeh, J.-M. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewéius, R. Yang, G. Welch, H. Towles, D. Nisté, and M. Pollefeys. Towards urban 3d reconstruction from video. In *3DPVT*, 2006. 2
- [2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded up robust features. *CVIU*, 110(3):346–359, 2008. 3
- [3] S. Blunsden. SFM-SIFT. http://homepages.inf.ed.ac.uk/s0346435/projects/sfm/sfm_sift.html (08-31-2007). 18, 23
- [4] A. Bruckstein, D. Donoho, and M. Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51(1):34–81, 2009. 9
- [5] E. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *arXiv*, 2009. 5
- [6] I. Esteban, J. Dijk, and F. Groen. Fit3d toolbox: multiple view geometry and 3d reconstruction for matlab. In *International Symposium on Security and Defence Europe (SPIE)*, 2010. 16, 18
- [7] J. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. Building Rome on a cloudless day. In *ECCV*, 2010. 2
- [8] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009. 3
- [9] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: image understanding using qualitative geometry and mechanics. In *ECCV*, 2010. 3
- [10] K. Huang, A. Yang, W. Hong, and Y. Ma. Large baseline matching and reconstruction from symmetry cells. In *ICRA*, 2004. 3
- [11] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 3
- [12] Y. Liu, H. Hel-Or, C. Kaplan, and L. van Gool. Computational symmetry in computer vision and computer graphics. *FTCGV*, 5:1–197, 2010. 3
- [13] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999. 3, 12, 13, 16, 18, 23
- [14] Y. Ma, J. Košecká, S. Soatto, and S. Sastry. *An Invitation to 3-D Vision, From Images to Models*. Springer-Verlag, New York, 2004. 5
- [15] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, pages 384–396, 2002. 3
- [16] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 65(1–2):43–72, 2005. 3
- [17] B. Mičušík and J. Košecká. Piecewise planar city 3D modeling from street view panoramic sequences. In *ICCV*, 2009. 2, 3

- [18] B. Mičušík, H. Wildenauer, and J. Košecká. Detection and matching of rectilinear structures. In *CVPR*, 2008. 3
- [19] M. Park, K. Brocklehurst, R. Collins, and Y. Liu. Deformed lattice detection in real-world images using mean-shift belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1), October 2009. 3, 11, 18, 19
- [20] S. Rao, H. Mobahi, A. Yang, S. Sastry, and Y. Ma. Natural image segmentation with adaptive texture and boundary encoding. In *ACCV*, 2009. 8, 10
- [21] A. Saxena, M. Sun, and A. Ng. Make3D: Learning 3D scene structure from a single still image. *PAMI*, 2009. 3
- [22] N. Snavely, S. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *IJCV*, 80:189–210, 2008. 2, 16, 18
- [23] P. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London*, 356(1740):1321–1340, 1998. 16
- [24] J. Čech, J. Matas, and M. Perdoch. Efficient sequential correspondence selection by cosegmentation. *PAMI*, 32(9):1568–1581, 2010. 8
- [25] A. Vedaldi and S. Soatto. Local features, all grown up. In *CVPR*, 2006. 8
- [26] Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. TILT: transform invariant low-rank textures. In *ACCV*, 2010. 3, 4, 5
- [27] S. Zhu and D. Mumford. A stochastic grammar of images. *FTCGV*, 2006. 3

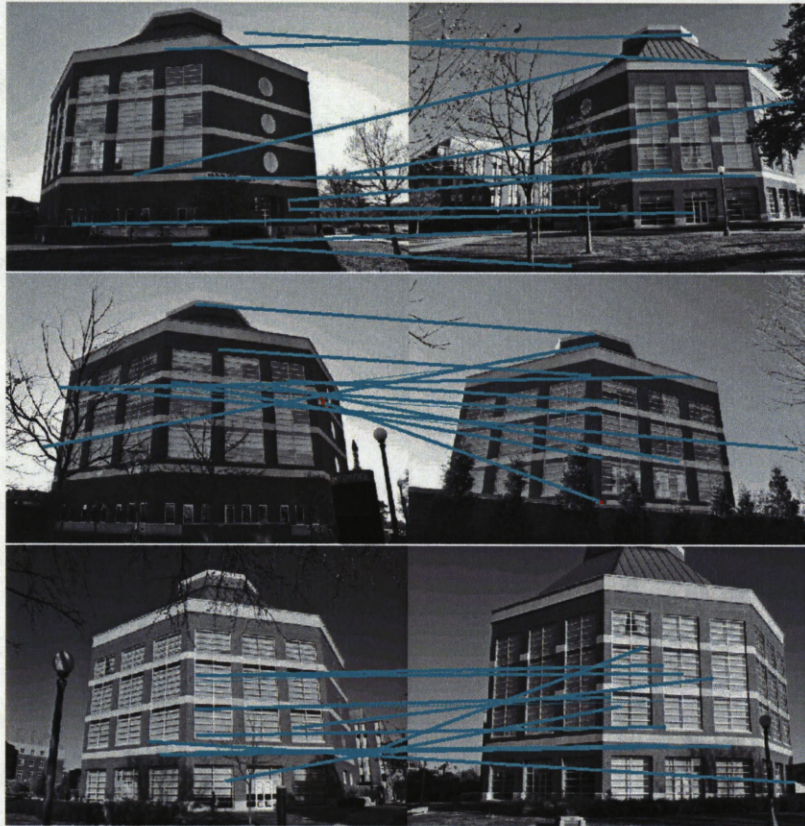


Figure 13: Feature-point matching by SIFT [13] from SFM-SIFT package [3] applied to consecutive image pairs from our octagonal building sequence. In each image, there are two dominant facades (and possibly little portions of other facades). The left dominant facade in the left images corresponds to the right dominant facade in the right images.



Figure 14: Voodoo camera tracker crashes on our data (top), while it works on the example sequence that comes with a package (bottom). The latter sequence is almost a video sequence, where most of the local features in each frame can be reliably tracked in the next frame. This is obviously not the case for our data where the two successive images are significantly different by any local measure.

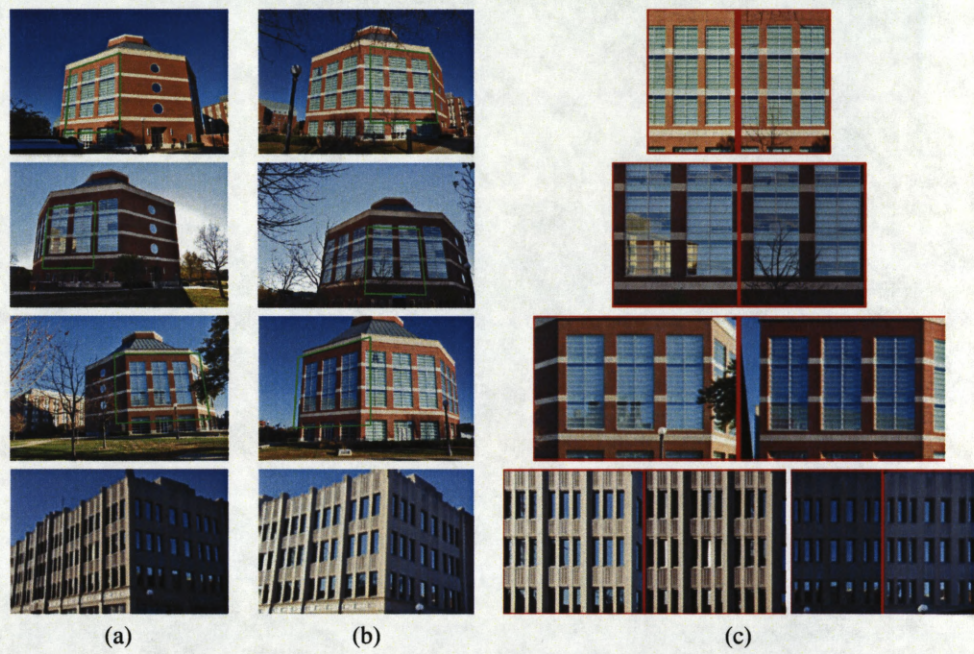


Figure 15: (a) and (b) show a pair of matched regions from the same facade with different partial occlusion. (c) shows the side by side and dense correspondence of the pixels.