Holistic 3D Reconstruction of Urban Structures from Low-Rank Textures

Hossein Mobahi[1], Zihan Zhou[2], Allen Y. Yang[3] and Yi Ma[2,4]
[1]CS Dept., University of Illinois at Urbana-Champaign
[2]ECE Dept., University of Illinois at Urbana-Champaign
[3]EECS Dept., University of California, Berkeley
[4]Visual Computing Group, Microsoft Research Asia
{hmobahi2,zzhou7}@illinois.edu, yang@eecs.berkeley.edu, mayi@microsoft.com

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 single or multiple images of the scene. We demonstrate how to construct 3D models of large-scale buildings from sequences of multiple large-baseline uncalibrated images that conventional SFM systems do not apply.

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. 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, 21, 16, 6].

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 [21, 6], more sophisticated techniques have been proposed to extract and match richer image features beyond points and edges. Examples include affine-invariant features (SIFT) [12, 14, 15, 2], superpixels [16], and object part-based regions [26, 7], 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 [11] for a review on this topic) or global structures such as vanishing points [17]. 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 [18, 11], 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 pur-

[1]Of 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.

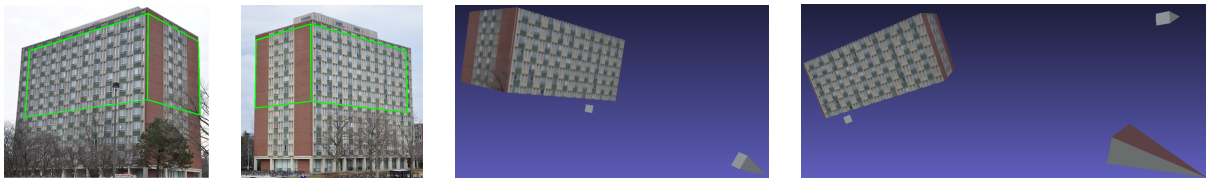

Figure 1. **Left Pair:** Example of matched facades of a building. **Right Pair:** Automatically reconstructed 3D model from *only four uncalibrated* images around the building by our method. Each image covers a pair of facades. The pyramids show the estimated location of cameras.

poses. 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 [9]. 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 [10, 8, 20]. In contrast, in this paper, we investigate a novel approach to infer accurate 3D geometry from either single or multiple images of an urban scene in a mainly unsupervised fashion.

To avoid the aforementioned difficulties while inferring 3D geometry, we exploit a new class of "building blocks" for modeling urban objects. These new tools 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. Motivated by a new type of image feature called transform invariant low-rank texture (TILT) [25], in this paper, we study 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 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 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.

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.

2. Geometry from One Facade of a Building

For completeness, we first give a brief review of work on low-rank textures [25] and then show how to use them for 3D modeling. It has been observed by the authors of [25] 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 [25] 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 [25] as:

$$I \circ \tau = I_0 + E, \tag{1}$$

where E represents some sparse error that corrupts the image, say due to partial occlusion. As shown in the work [25]

and Robust PCA literature [4], 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, respectively [2].

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: $\tau^{-1} \doteq [\boldsymbol{t}_1, \boldsymbol{t}_2, \boldsymbol{t}_3] = K[\boldsymbol{n}_1, \boldsymbol{n}_2, T]$, where $R = [\boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3] \in \mathbb{R}^{3\times3}$ is the rotation, $T \in \mathbb{R}^3$ the translation, and $K \in \mathbb{R}^{3\times3}$ 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 $\boldsymbol{v}_1 = \boldsymbol{t}_1, \boldsymbol{v}_2 = \boldsymbol{t}_2 \in \mathbb{R}^3$ in the image coordinates, respectively [13]. 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 $K \in \mathbb{R}^{3\times3}$: $\boldsymbol{v}_1^T K^{-T} K^{-1} \boldsymbol{v}_2 = 0$. 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 $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3 \in \mathbb{R}^3$, respectively. Notice that the pairs $(\boldsymbol{v}_1, \boldsymbol{v}_3)$ and $(\boldsymbol{v}_2, \boldsymbol{v}_3)$ 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 that go through the vanishing point $\boldsymbol{v}_3$ 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.

[2]The 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 $\|E\|_1 = \sum_{i,j} |e_{ij}|$.

(a) (b) (c)

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).

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 $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively, as shown in Figure 3(a). As one can see, since the vanishing points $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3$ are known, any intersection line l between $\boldsymbol{A}$ and $\boldsymbol{B}$ 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 $\boldsymbol{v}_3$. 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:

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

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

If the camera is calibrated, from the assumption we know $\boldsymbol{v}_3$ should be orthogonal to both $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ as $\boldsymbol{v}_3 \sim \boldsymbol{v}_1 \times \boldsymbol{v}_2$. Very often, the two facades of the building are also

(a) (b) (c) (d)

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.

orthogonal to each other, i.e., $\boldsymbol{v}_1 \perp \boldsymbol{v}_2$.[3] If the camera is not calibrated, the three vanishing points impose three independent constraints on the camera intrinsic parameters:

$$\boldsymbol{v}_1^T K^{-T} K^{-1} \boldsymbol{v}_2 = 0, \boldsymbol{v}_1^T K^{-T} K^{-1} \boldsymbol{v}_3 = 0, \boldsymbol{v}_2^T K^{-T} K^{-1} \boldsymbol{v}_3 = 0.$$

This allows us to fully calibrate the camera from just a pair of intersecting facades, 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.

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* [24, 23]. 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 length[4], 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 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 tolerance:

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

where $size(I)$ 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

[3] This may not always be the case. For instance, the facades in Figure 9 (a) and (b) are not orthogonal.

[4] There is a theoretical connection between rank and the coding length of a matrix [19]. 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.

[5] We 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.

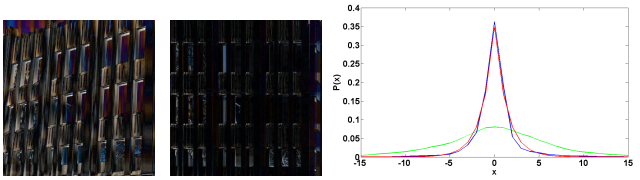

Figure 5. **Left:** The residual matrix $\hat{E}$ of the original (left) and transformed (right) images in Figure 6 approximated by top three singular values/vectors. **Right**: Empirical probability distributions of the errors for the left (green) and right (blue) residual maps. The empirical distribution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

that most sparse signals obey a *Laplace distribution* [3]: $p(x) = \frac{1}{2\lambda}\exp\big(-\frac{|x-\mu|}{\lambda}\big)$, where we typically assume $\mu = 0$ in our setting. Since we here are working with a set of discrete samples: $\mathcal{X} = \{x_1, \ldots, x_N\}$[6], we can work with a discrete Laplace distribution $p_k = \frac{1}{Z\Lambda}\exp\big(-\frac{|x_k|}{\Lambda}\big)$, 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:

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

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: $\hat{A} = \sum_{i=1}^{r}\hat{\sigma}_i\hat{\boldsymbol{u}}_i\hat{\boldsymbol{v}}_i^T$, where the non-quantized variables are $\boldsymbol{u}_i \in \mathbb{R}^m$, $\boldsymbol{v}_i \in \mathbb{R}^n$ and $\sigma_i \in \mathbb{R}_+$. Obviously, for $r \ll \min\{m, n\}$, this encoding uses much fewer bits than the naive encoding $(m+n+1)rf \ll mnf$.

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

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

We can further compress the vectors $\{\hat{\boldsymbol{u}}_i, \hat{\boldsymbol{v}}_i\}$ based on additional structures in them. As each $\hat{\boldsymbol{u}}_i$ or $\hat{\boldsymbol{v}}_i$ 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.

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

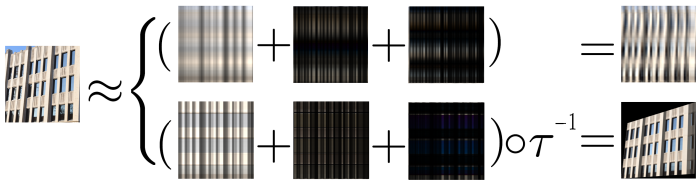

Figure 6. Approximation of a facade image with the top three singular components $\sum_{i=1}^{3}\sigma_i\boldsymbol{u}_i\boldsymbol{v}_i^T$. **Top:** SVD of the original image. **Bottom:** SVD of the rectified image by TILT.

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:

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

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:

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

We solve this problem in a greedy and agglomerative fashion, similar to that in [19]. 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.

Comparison with Symmetry Detection. Conceptually, one could also utilize the work of [18] 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. These problems make the results hard to use for subsequent matching.

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

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

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).

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 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:

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

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 8 (e), we illustrate the difficulty of applying the classical SIFT matching technique [12] 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 9. 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 9(a) and (b). We further arrange the sequence of images so that matching of common facades is only performed between consecutive images. See Figure 9(c) and (d) for an example of the matched facades.

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

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

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

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. (d) Point-wise match between two regions of the facades. (e) Feature-point matching result of the two rectified regions by SIFT [12], with red lines indicate mismatches.

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

Figure 9. **(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.

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 9(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 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 9(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 n-sided polygon on the $y = 0$ plane. For example, the 3D model of the octagonal building ($n = 8$) is shown in Figure 9(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 $(A_i^j, E_i^j), 1 \le j \le 2$: $I_i \circ \tau_i^j(K_i, R_i, T_i, X) = A_i^j + E_i^j$. Second, we ask the i-th pair of matching facades to be the same, up to some sparse error $\boldsymbol{e}_i$:

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

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

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

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 $(K_i, R_i, T_i, X_i)_{i=1}^n$. 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 10 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 9(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 [21], SFM-SIFT [10] (which combines Torr's SFM toolbox [22] with SIFT feature detector [12]), FIT3D

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

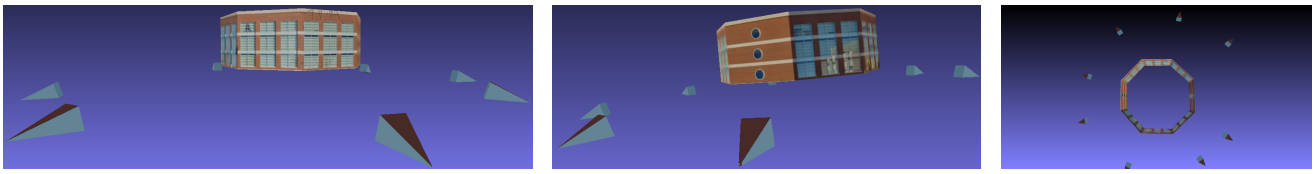

Figure 10. Frontal (left & middle) and top (right) views of the recovered building. Each pyramid shows the estimated location of a camera.

[5], and Voodoo Camera Tracker [11]. All these packages report errors related to their inability of establishing meaningful correspondence across the views.

7. Acknowledgement

This research was in part supported by ARO MURI W911NF-06-1-0076, ARL MAST-CTA W911NF-08-2-0004, ONR N00014-09-1-0230, NSF CCF 09-64215, NSF IIS 11-16012 and DARPA KECoM 10036-100471. Hossein Mobahi is supported by CSE Ph.D fellowship of UIUC. 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.

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. 1

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded up robust features. *CVIU*, 110(3):346–359, 2008. 1

[3] 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. 5

[4] E. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *arXiv*, 2009. 3

[5] 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. 8

[6] 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. 1

[7] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009. 1

[8] A. Gupta, A. Efros, and M. Hebert. Blocks world revisited: image understanding using qualitative geometry and mechanics. In *ECCV*, 2010. 2

[9] K. Huang, A. Yang, W. Hong, and Y. Ma. Large baseline matching and reconstruction from symmetry cells. In *ICRA*, 2004. 2

[10] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 2

[11] Y. Liu, H. Hel-Or, C. Kaplan, and L. van Gool. Computational symmetry in computer vision and computer graphics. *FTCGV*, 5:1–197, 2010. 1

[12] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999. 1, 6, 7

[13] Y. Ma, J. Košecká, S. Soatto, and S. Sastry. *An Invitation to 3-D Vision, From Images to Models*. Springer-Verlag, New York, 2004. 3

[14] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, pages 384–396, 2002. 1

[15] 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. 1

[16] B. Mičušík and J. Košecká. Piecewise planar city 3D modeling from street view panaramic sequences. In *ICCV*, 2009. 1

[17] B. Mičušík, H. Wildenauer, and J. Košecká. Detection and matching of rectilinear structures. In *CVPR*, 2008. 1

[18] 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. 1, 5

[19] S. Rao, H. Mobahi, A. Yang, S. Sastry, and Y. Ma. Natural image segmentation with adaptive texture and boundary encoding. In *ACCV*, 2009. 4, 5

[20] A. Saxena, M. Sun, and A. Ng. Make3D: Learning 3D scene structure from a single still image. *PAMI*, 2009. 2

[21] N. Snavely, S. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *IJCV*, 80:189–210, 2008. 1, 7

[22] P. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London*, 356(1740):1321–1340, 1998. 7

[23] J. Čech, J. Matas, and M. Perdoch. Efficient sequential correspondence selection by cosegmentation. *PAMI*, 32(9):1568–1581, 2010. 4

[24] A. Vedaldi and S. Soatto. Local features, all grown up. In *CVPR*, 2006. 4

[25] Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. TILT: transform invariant low-rank textures. In *ACCV*, 2010. 2

[26] S. Zhu and D. Mumford. A stochastic grammar of images. *FTCGV*, 2006. 1

[11] http://www.digilab.uni-hannover.de/docs/manual.html