Multiview Geometry for Texture Mapping 2D Images Onto 3D Range Data *

Computer Vision and Pattern Recognition, 2006

Lingyun Liu and Ioannis Stamos
Dept. of Computer Science
Hunter College / CUNY
New York, NY 10021
istamos@hunter.cuny.edu

Gene Yu and George Wolberg
Dept. of Computer Science
City College / CUNY
New York, NY 10031
wolberg@cs.ccny.cuny.edu

Siavash Zokai
Brainstorm Technology
New York, NY 10011
zokai@brainstormllc.com

Abstract

The photorealistic modeling of large-scale scenes, such as urban structures, requires a fusion of range sensing technology and traditional digital photography. This paper presents a system that integrates multiview geometry and automated 3D registration techniques for texture mapping 2D images onto 3D range data. The 3D range scans and the 2D photographs are respectively used to generate a pair of 3D models of the scene. The first model consists of a dense 3D point cloud, produced by using a 3D-to-3D registration method that matches 3D lines in the range images. The second model consists of a sparse 3D point cloud, produced by applying a multiview geometry (structure-from-motion) algorithm directly on a sequence of 2D photographs. This paper introduces a novel algorithm for automatically recovering the rotation, scale, and translation that best aligns the dense and sparse models. This alignment is necessary to enable the photographs to be optimally texture mapped onto the dense model. The contribution of this work is that it merges the benefits of multiview geometry with automated registration of 3D range scans to produce photorealistic models with minimal human interaction. We present results from experiments in large-scale urban scenes.

1. Introduction

The photorealistic modeling of large-scale scenes, such as urban structures, requires a combination of range sensing technology with traditional digital photography. A systematic way for registering 3D range scans and 2D images is thus essential. This paper presents a system that integrates multiview geometry and automated 3D registration techniques for texture mapping 2D images onto 3D range data. The novelty of our approach is that it exploits all possible relationships between 3D range scans and 2D images by performing 3D-to-3D range registration, 2D-to-3D image-to-range registration, and structure from motion. Several papers, including this one, provide frameworks for automated texture mapping onto 3D range scans [15, 19, 29, 33, 38]. These methods are based on extracting features (e.g., points, lines, edges, rectangles or rectangular parallelepipeds) and matching them between the 2D images and the 3D range scans. Our approach provides a solution of increased robustness, efficiency and generality with respect to previous methods. Our contribution is discussed in Sec. 2.

Despite the advantages of feature-based texture mapping solutions, most systems that attempt to recreate photorealistic models do so by requiring the manual selection of features among the 2D images and the 3D range scans, or by rigidly attaching a camera onto the range scanner and thereby fixing the relative position and orientation of the two sensors with respect to each other [1, 9, 25, 28, 37]. The fixed-relative position approach provides a solution that has the following major limitations:

1. The acquisition of the images and range scans occur at the same point in time and from the same location in space. This leads to a lack of 2D sensing flexibility since the limitations of 3D range sensor positioning, such as standoff distance and maximum distance, will cause constraints on the placement of the camera. Also, the images may need to be captured at different times, particularly if there were poor lighting conditions at the time that the range scans were acquired.

2. The static arrangement of 3D and 2D sensors prevents the camera from being dynamically adjusted to the requirements of each particular scene. As a result, the focal length and relative position must remain fixed.

3. The fixed-relative position approach cannot handle the case of mapping historical photographs on the models

*Supported in part by NSF CAREER IIS-01-21239, NSF MRI/RUI EIA-0215962, ONR N000140310511, and NIST ATP 70NANB3H3056.

or of mapping images captured at different instances in time. These are capabilities that our method achieves.

In summary, fixing the relative position between the 3D range and 2D image sensors sacrifices the flexibility of 2D image capture. Alternatively, methods that require manual interaction for the selection of matching features among the 3D scans and the 2D images are error-prone, slow, and not scalable to large datasets. These limitations motivate the work described in this paper, making it essential for producing photorealistic models of large-scale urban scenes.

The texture mapping solution described in this paper merges the benefits of multiview geometry with automated 3D-to-3D range registration and 2D-to-3D image-to-range registration to produce photorealistic models with minimal human interaction. The 3D range scans and the 2D photographs are respectively used to generate a pair of 3D models of the scene. The first model consists of a dense 3D point cloud, produced by using a 3D-to-3D registration method that matches 3D lines in the range images to bring them into a common reference frame. The second model consists of a sparse 3D point cloud, produced by applying a multiview geometry (structure-from-motion) algorithm directly on a sequence of 2D photographs to simultaneously recover the camera motion and the 3D positions of image features. This paper introduces a novel algorithm for automatically recovering the similarity transformation (rotation/scale/translation) that best aligns the sparse and dense models. This alignment is necessary to enable the photographs to be optimally texture mapped onto the dense model. No a priori knowledge about the camera poses relative to the 3D sensor's coordinate system is needed, other than the fact that one image frame should overlap the 3D structure (see Sec. 4). Given one sparse point cloud derived from the photographs and one dense point cloud produced by the range scanner, a similarity transformation between the two point clouds is computed in an automatic and efficient way (Fig. 1). The framework of our system is:

- A set of 3D range scans of the scene are acquired and co-registered to produce a dense 3D point cloud in a common reference frame (Sec. 3).
- An independent sequence of 2D images is gathered, taken from various viewpoints that do not necessarily coincide with those of the range scanner. A sparse 3D point cloud is reconstructed from these images by using a structure-from-motion (SFM) algorithm (Sec. 5).
- A *subset* of the 2D images are automatically registered with the dense 3D point cloud acquired from the range scanner (Sec. 4).
- Finally, the *complete* set of 2D images is automatically aligned with the dense 3D point cloud (Sec. 6). This last step provides an integration of all the 2D and 3D data in the same frame of reference. It also provides the transformation that aligns the models gathered via range sensing and computed via structure from motion.

2. Related Work

There are many approaches for the solution of the pose estimation problem from both point correspondences [22, 26] and line correspondences [6, 13], when a set of matched 3D and 2D points or lines are known, respectively. In the early work of [8], the probabilistic RANSAC method was introduced for automatically computing matching 3D and 2D points. This approach works well only when the percentage of incorrectly matched pairs (outliers) is small. Solutions in automated matching of 3D with 2D features in the context of object recognition and localization include [4, 11, 14, 16, 17, 35]. Very few methods, though, attack the problem of automated alignment of images with dense point clouds derived from range scanners. This problem is of major importance for automated photorealistic reconstruction of large-scale scenes from range and image data. In [29, 19] two methods that exploit orthogonality constraints (rectangular features and vanishing points) in man-made scenes are presented. The methods can provide excellent results, but will fail in the absence of a sufficient number of linear features. Ikeuchi [15], on the other hand, presents an automated 2D-to-3D registration method that relies on the reflectance range image. However, the algorithm requires an initial estimate of the image-to-range alignment in order to converge. Finally, [33] presents a method that works under specific outdoor lighting situations.

A system whose goals are very similar to ours is described in [38]. In that work, continuous video is aligned onto a 3D point cloud obtained from a 3D sensor. First, an SFM/stereo algorithm produces a 3D point cloud from the video sequence. This point cloud is then registered to the 3D point cloud acquired from the range scanner by applying the ICP algorithm [3]. One limitation of this approach has to do with the shortcomings of the ICP algorithm. In particular, the 3D point clouds must be manually brought close to each to yield a good initial estimate that is required for the ICP algorithm to work. The ICP may fail in scenes with few discontinuities, such as those replete with planar or cylindrical structures. Also, in order for the ICP algorithm to work, a very dense model from the video sequence must be generated. This means that the method of [38] is restricted to video sequences, which limits the resolution of the 2D imagery. Finally, that method does not automatically compute the difference in scale between the range model and the recovered SFM/stereo model.

Our contributions can be summarized as follows:

- Like [38], we compute a model from a collection of

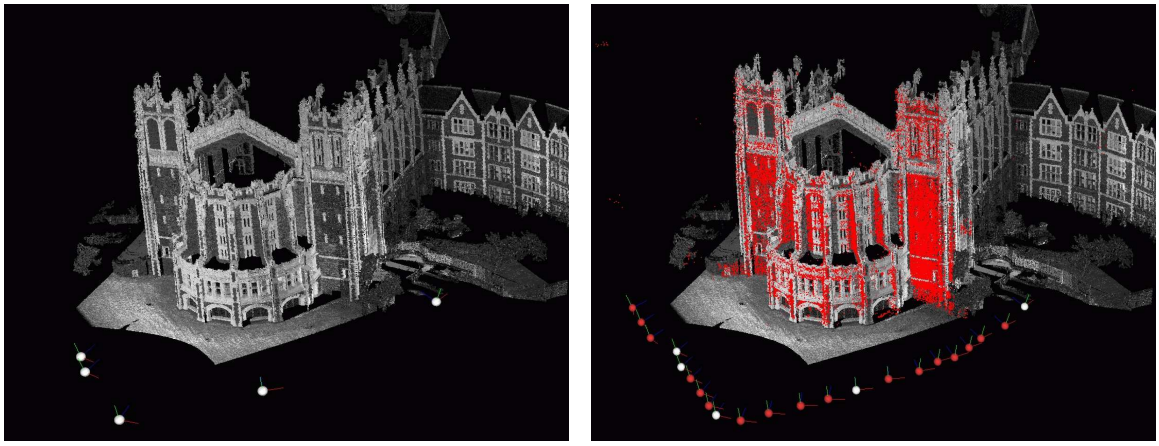


Figure 1. **(a)**: 22 registered range scans of Shepard Hall (CCNY) constitute a dense 3D point cloud model M_{range} (Sec. 3). The color of each 3D point corresponds to the intensity of the returned laser beam, and no texture mapping has been applied yet. The five white dots correspond to the locations of the 2D images that are independently registered with the model M_{range} via a 2D-to-3D image-to-range registration algorithm (Sec. 4). **(b)**: The 3D range model M_{range} overlaid with the 3D model M_{sfm} produced by SFM (Sec. 5) after the alignment method of Sec. 6. The points of M_{sfm} are shown in red, and the sequence of 2D images that produced M_{sfm} are shown as red dots in the figure. Their positions have been accurately recovered with respect to both models M_{range} and M_{sfm} (Sec. 6).

images via SFM. Our method for aligning the range and SFM models, described in Sec. 6, does not rely on ICP and thus does not suffer from its limitations.

- We are able to automatically compute the scale difference between the range and SFM models.
- Like [19], we perform 2D-to-3D image-to-range registration for a few (at least one) images of our collection. This feature-based method provides excellent results in the presence of a sufficient number of linear features. Therefore, the images that contain enough linear features are registered using that method. The utilization of the SFM model allows us to align the remaining images with a method that involves robust point (and not line) correspondences.
- We generate an optimal texture mapping result by using contributions of all 2D images.

3. 3D-to-3D Range Registration

The first step is to acquire a set of range scans $R_m (m = 1, \ldots, M)$ that adequately covers the 3D scene. The laser range scanner used in our work is a Cyrax 2500 [18], an active sensor that sweeps an eye-safe laser beam across the scene. It is capable of gathering one million 3D points at a maximum distance of 100 meters with an accuracy of 5mm. Each 3D point is associated with four values $(x, y, z, l)^T$, where $(x, y, z)^T$ is its Cartesian coordinates in the scanner's local coordinate system, and l is the laser intensity of the returned laser beam.

Each range scan then passes through an automated segmentation algorithm [30] to extract a set of major 3D planes and a set of geometric 3D lines G_i from each scan $i = 1, \ldots, M$. The geometric 3D lines are computed as the intersections of segmented planar regions and as the borders of the segmented planar regions. In addition to the geometric lines G_i, a set of reflectance 3D lines L_i are extracted from each 3D range scan. The range scans are registered in the same coordinate system via the automated 3D-to-3D feature-based range-scan registration method of [5, 31]. The method is based on an automated matching procedure of linear features of overlapping scans. As a result, all range scans are registered with respect to one selected pivot scan. The set of registered 3D points from the M scans is called M_{range} (Fig. 1(a)).

4. 2D-to-3D Image-to-Range Registration

The automated 2D-to-3D image-to-range registration method of [19] is used for the automated calibration and registration of a single 2D image I_n with the 3D range model M_{range}. The computation of the rotational transformation between I_n and M_{range} is achieved by matching at least two vanishing points computed from I_n with major scene directions computed from clustering the linear features extracted from M_{range}. The method is based on the

assumption that the 3D scene contains a cluster of vertical and horizontal lines. This is a valid assumption in urban scene settings.

The internal camera parameters consist of focal length, principal point, and other parameters in the camera calibration matrix $\mathcal{K}$ [10]. They are derived from the scene's vanishing points, whereby the 2D images are assumed to be free of distortion. Finally, the translation between I_n and M_{range} is computed after higher-order features such as 2D rectangles from the 2D image and 3D parallelepipeds from the 3D model are extracted and automatically matched.

With this method, a few 2D images can be independently registered with the model M_{range}. The algorithm will fail to produce satisfactory results in parts of the scene where there is a lack of 2D and 3D features for matching. Also, since each 2D image is independently registered with the 3D model, valuable information that can be extracted from relationships between the 2D images (SFM) is not utilized. In order to solve the aforementioned problems, an SFM module (Sec. 5) and final alignment module (Sec. 6) has been added into the system. These two modules increase the robustness of the reconstructed model, and improve the accuracy of the final texture mapping results. Therefore, the 2D-to-3D image-to-range registration algorithm is used in order to register a few 2D images (five shown in Fig. 1(a)) that produce results of high quality. The final registration of the 2D image sequence with the range model M_{range} is performed after SFM is utilized (Sec. 5).

5. Multiview pose estimation and 3D structure reconstruction

The input to our system is a sequence $\mathbf{I} = \{I_n | n = 1, \ldots, N\}$ of high resolution still images that capture the 3D scene. This is necessary to produce photorealistic scene representations. Therefore we have to attack the problem of finding correspondences in a sequence of wide-baseline high resolution images, a problem that is much harder than feature tracking from a video sequence. Fortunately, there are several recent approaches that attack the wide-baseline matching problem [27, 34, 20]. For the purposes of our system, we have adopted the scale-invariant feature transform (SIFT) method [20] for pairwise feature extraction and matching. In general, structure from motion (SFM) from a set images has been rigorously studied [7, 10, 21].

Our method for pose estimation and partial structure recovery is based on sequential updating. The method is similar to work explained in [2, 24]. In order to get very accurate pose estimation, we assume that the camera(s) are precalibrated. It is, of course, possible to recover unknown and varying focal length by first recovering pose and structure up to an unknown projective transform and then upgrading to Euclidean space as shown in [12, 32, 23]. However, some of the assumptions that these methods make (e.g., no skew, approximate knowledge of the aspect ratio and principal point) may produce visible mismatches in a high resolution texture map. Thus, for the sake of accuracy we are utilizing the camera calibration method of [36].

The following steps describe our SFM implementation. First, we determine the lens distortion and compensate for it in images I_i $i = 1, \cdots, N$. Then, for each pair i and $i+1$, a list of 2D feature matches is generated using SIFT [20]. An initial motion and structure is computed from the first two images I_1 and I_2 as follows. The relative pose (rotation R, and translation T) is calculated by the decomposition of the essential matrix $E = \mathcal{K}^T F \mathcal{K}$, after the fundamental matrix F computation (via RANSAC to eliminate outliers). The matrix $\mathcal{K}$ contains the internal camera calibration parameters. The pose of the first camera (I_1) is set to $R_1 = \mathbf{I}, T_1 = \mathbf{0}$, and for the second ($I_2$) to $R_2 = R, T_2 = T$. Then, an initial point cloud of 3D points $\mathbf{X_j}$ is computed from the 2D correspondences between I_1 and I_2 through triangulation. Finally, the relative pose and 3D structure is refined via the minimization of the following meaningful geometric reprojection error: $\min_{R_i, T_i, \mathbf{X_j}} \sum_{i=1}^{2} \sum_{j} \|m_{ij} - \mathcal{K}[R_i \mid T_i]\mathbf{X_j})\|^2$, where (m_{1j}, m_{2j}) is the pair of matching 2D features between images I_1 and I_2 that produced the point $\mathbf{X_j}$.

After the initial motion and structure is computed from first pair, the remaining pairs are used to further augment the SFM computation. For each image $I_i, i = 3 \cdots N$ the following operations are performed:

- A set of common features are found between the three images I_{i-2}, I_{i-1}, and I_i. These are features that have been tracked from frame I_{i-2} to frame I_{i-1} and then to frame I_i via the SIFT algorithm. The 3D points associated with the matched features between I_{i-2} and I_{i-1} are recorded as well.

- From the 2D features and 3D points collected in the previous step, the pose (R_i, T_i) of image I_i is computed using the Direct Linear Transform (DLT) with RANSAC for outlier detection. Finally, the pose is further refined via a nonlinear steepest-descent algorithm.

- A new set of 3D points $\mathbf{X'_j}$ can now be computed from the remaining 2D features that are seen only in images I_{i-1} and I_i (these features where not seen in image I_{i-2} and thus no 3D point was computed for them). These new 3D points are projected onto the previous images of the sequence $I_{i-2}, ..., I_1$ in order to reinforce more correspondences (normalized correlation with subpixel accuracy) between sub-sequences of the images in the list.

- Finally, these new (corresponding) features and 3D points $\mathbf{X'_j}$ are added to the database of feature correspondences/3D points. Tests that detect duplicate features and occlusions occur before their addition to the database.

The final step is the refinement of the computed pose and structure by a global bundle adjustment procedure that involves all images of the sequence. In order to do that we are using 2D feature points that are either fully or partially tracked throughout the sequence. This procedure minimizes the following reprojection error:

$$\min_{R_i,T_i,\mathbf{X_j}} \sum_{i=1}^{N} \sum_{j} \|m_{ij} - \mathcal{K}[R_i \mid T_i]\mathbf{X_j})\|^2$$

In the previous formula each sequence of tracked 2D feature points $(m_{1j}, m_{2j}, \ldots, m_{nj})$ correspond to the reconstructed 3D point $\mathbf{X_j}$.

6. Alignment of 2D Image Sequences Onto 3D-Range Point Clouds

The set of dense range scans $\{R_m | m = 1, \ldots, M\}$ are registered in the same reference frame (Sec. 3), producing a 3D range model called M_{range}. On the other hand, the sequence of 2D images $\mathbf{I} = \{I_n | n = 1, \ldots, N\}$ produces a sparser 3D model of the scene (Sec. 5) called M_{sfm}. Both of these models are represented as clouds of 3D points. The distance between any two points in M_{range} corresponds to the actual distance of the points in 3D space, whereas the distance of any two points in M_{sfm} is the actual distance multiplied by an unknown scale factor s. In order to align the two models a similarity transformation that includes the scale factor s, a rotation R and a translation T needs to be computed. In this section, a novel algorithm that automatically computes this transformation is presented. The transformation allows for the optimal texture mapping of all images onto the dense M_{range} model, and thus provides photorealistic results of high quality.

Every point X from M_{sfm} can be projected onto a 2D image $I_n \in \mathbf{I}$ by the following transformation:

$$\mathbf{x} = \mathcal{K}_n[R_n \,|\, T_n\,]\,\mathbf{X} \tag{1}$$

where $\mathbf{x} = (x, y, 1)$ is a pixel on image I_n, $\mathbf{X} = (X, Y, Z, 1)$ is a point of M_{sfm}, $\mathcal{K}_n$ is the projection matrix, R_n is the rotation transformation and T_n is the translation vector. These matrices and points $\mathbf{X}$ are computed by the SFM method (Sec. 5).

Some of the 2D images $\mathbf{I'} \subset \mathbf{I}$ are also automatically registered with the 3D range model M_{range} (Sec. 4). Thus, each point of M_{range} can be projected onto each 2D image $I_n \in \mathbf{I'}$ by the following transformation:

$$\mathbf{y} = \mathcal{K}_n[R'_n \,|\, T'_n\,]\,\mathbf{Y} \tag{2}$$

where $\mathbf{y} = (x, y, 1)$ is a pixel in image I_n, $\mathbf{Y} = (X, Y, Z, 1)$ is a point of model M_{range}, $\mathcal{K}_n$ is the projection matrix of I_n, R'_n is the rotation and T'_n is the translation. These transformations are computed by the 2D-to-3D registration method (Sec. 4).

The key idea is to use the images in $I_n \in \mathbf{I'}$ as references in order to find the corresponding points between M_{range} and M_{sfm}. The similarity transformation between M_{range} and M_{sfm} is then computed based on these correspondences. In summary, the algorithm works as follows:

1. Each point of M_{sfm} is projected onto $I_n \in \mathbf{I'}$ using Eq. (1). Each pixel $p_{(i,j)}$ of I_n is associated with the closest projected point $\mathbf{X} \in M_{sfm}$ in an $L \times L$ neighborhood on the image. Each point of M_{range} is also projected onto I_n using Eq. (2). Similarly, each pixel $p_{(i,j)}$ is associated with the projected point $\mathbf{Y} \in M_{range}$ in an $L \times L$ neighborhood (Fig. 2). Z-buffering is used to handle occlusions.

2. If a pixel $p_{(i,j)}$ of image I_n is associated with a pair of 3D points $(\mathbf{X}, \mathbf{Y})$, one from M_{sfm} and the other from M_{range}, then these two 3D points are considered as candidate matches. Thus, for each 2D-image in $\mathbf{I'}$ a set of matches is computed, producing a collection of candidate matches named $\mathbf{L}$. These 3D-3D correspondences between points of M_{range} and points of M_{sfm} could be potentially used for the computation of the similarity transformation between the two models. The set $\mathbf{L}$ contains many outliers, due to the very simple closest-point algorithm utilized. However, $\mathbf{L}$ can be further refined (Sec. 6.1) into a set of robust 3D point correspondences $\mathcal{C} \subset \mathbf{L}$.

3. Finally, the transformation between M_{range} and M_{sfm} is computed by minimizing a weighted error function E (Sec. 6.1) based on the final robust set of correspondences $\mathcal{C}$.

6.1 Correspondence Refinement and Optimization

The set of candidate matches $\mathbf{L}$ computed in the second step of the previous algorithm contains outliers due to errors introduced from the various modules of the system (SFM, 2D-to-3D registration, range sensing). It is thus important to filter out as many outliers as possible through verification procedures. A natural verification procedure involves the difference in scale between the two models. Consider two pairs of plausible matched 3D-points $(\mathbf{X_1}, \mathbf{Y_1})$ and $(\mathbf{X_2}, \mathbf{Y_2})$ ($\mathbf{X_i}$ denotes points from the M_{sfm} model, while $\mathbf{Y_j}$ points from the the M_{range} model). If these were indeed correct correspondences, then the scale factor between between the two models would be $s = \|\mathbf{X_1} - \mathbf{X_2}\| / \|\mathbf{Y_1} -$

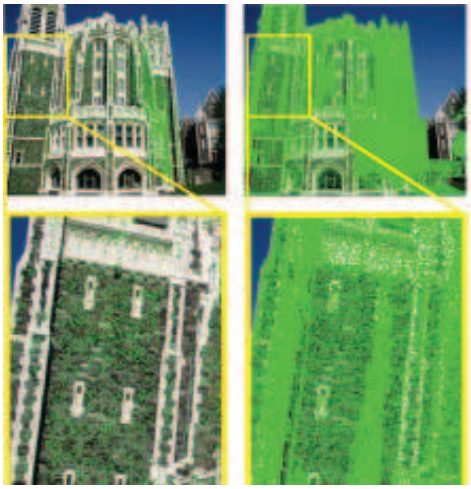


Figure 2. **(a)**: The points of model M_{sfm} projected onto one 2D image I_n (Sec. 5). The projected points are shown in green. **(b)**: The points of model M_{range} projected onto the same 2D image I_n (projected points shown in green) after the automatic 2D-to-3D registration (Sec. 4). Note that the density of 3D range points is much higher than the density of the SFM points, due to the different nature of the two reconstruction processes. Finding corresponding points between M_{range} and M_{sfm} is possible on the 2D image space of I_n. This yields the transformation between the two models (Sec. 6).

$\mathbf{Y_2}\|$. Since the computed scale factor should be the same no matter which correct matching pair is used, then a robust set of correspondences from $\mathbf{L}$ should contain only these pairs that produce the same scale factor s. The constant scale factor among correctly picked pairs is thus an invariant feature that we exploit. We now explain how we achieve this robust set of correspondences.

For each image $I_n \in \mathbf{I}'$, let us call the camera's center of projection as $\mathbf{C_n^{sfm}}$ in the local coordinate system of M_{sfm} and $\mathbf{C_n^{rng}}$ in the coordinate system of M_{range}. These two centers have been computed from two independent processes: SFM (Sec. 5) and 2D-to-3D registration (Sec. 4). Then for any candidate match, $(\mathbf{X}, \mathbf{Y}) \in \mathbf{L}$, a candidate scale factor $s_1(\mathbf{X}, \mathbf{Y})$ can be computed as:

$$s_1(\mathbf{X}, \mathbf{Y}) = \frac{\|\mathbf{X} - \mathbf{C_n^{sfm}}\|}{\|\mathbf{Y} - \mathbf{C_n^{rng}}\|}$$

If we keep the match $(\mathbf{X}, \mathbf{Y})$ fixed and we consider every other match $(\mathbf{X}', \mathbf{Y}') \in \mathbf{L}$, $L-1$ candidate scale factors $s_2(\mathbf{X}', \mathbf{Y}')$ and $L-1$ candidate scale factors $s_3(\mathbf{X}', \mathbf{Y}')$ (L is the number of matches in $\mathbf{L}$) are computed as:

$$s_2(\mathbf{X}', \mathbf{Y}') = \frac{\|\mathbf{X}' - \mathbf{C_n^{sfm}}\|}{\|\mathbf{Y}' - \mathbf{C_n^{rng}}\|}, s_3(\mathbf{X}', \mathbf{Y}') = \frac{\|\mathbf{X} - \mathbf{X}'\|}{\|\mathbf{Y} - \mathbf{Y}'\|}$$

That means that if we keep the match $(\mathbf{X}, \mathbf{Y})$ fixed, and consider all other matches $(\mathbf{X}', \mathbf{Y}')$ we can compute a triple of candidate scale factors: $s_1(\mathbf{X}, \mathbf{Y})$, $s_2(\mathbf{X}', \mathbf{Y}')$, and $s_3(\mathbf{X}', \mathbf{Y}')$. We then consider the two pairs of matches $(\mathbf{X}, \mathbf{Y})$ and $(\mathbf{X}', \mathbf{Y}')$ as *compatible* if the scale factors in the above triple are close with respect to each other. By fixing $(\mathbf{X}, \mathbf{Y})$, all matches that are compatible with it are found. The confidence in the match $(\mathbf{X}, \mathbf{Y})$ is the number of compatible matches it has. By going through all matches in $\mathbf{L}$, their confidence is computed via the above procedure. Out of these matches the one with the highest confidence is selected as the most prominent: $(\mathbf{X_p}, \mathbf{Y_p})$. Let us call $\mathbf{L_n}$ the set that contains $(\mathbf{X_p}, \mathbf{Y_p})$ and all other matches that are compatible with it. Note that this set is based on the centers of projection of image I_n as computed by SFM and 2D-to-3D registration. Let us also call s_n the scale factor that corresponds to the set $\mathbf{L_n}$. This scale factor can be computed by averaging the triples of scale factors of the elements in $\mathbf{L_n}$. Finally a different set $\mathbf{L_n}$ and scale factor s_n is computed for every image $I_n \in \mathbf{I}'$.

From the previous discussion it is clear that each $\mathbf{L_n}$ is a set of matches that is based on the center of projection of each image I_n independently. A set of matches that will provide a globally optimal solution should consider all images of $\mathbf{I}'$ simultaneously. Out of the scale factors computed from each set $\mathbf{L_n}$, the one that corresponds to the largest number of matches is the one more robustly extracted by the above procedure. That computed scale factor, s_{opt}, is used as the final filtration for the production of the robust set of matches $\mathcal{C}$ out of $\mathbf{L}$. In particular, for each candidate match $(\mathbf{X}, \mathbf{Y}) \in \mathbf{L}$, a set of scale factors are computed as

$$s'_n = \frac{\|\mathbf{X} - \mathbf{C_n^{sfm}}\|}{\|\mathbf{Y} - \mathbf{C_n^{rng}}\|}$$

where $n = 1, 2, ..., K$, and K is the number of images in $\mathbf{I}'$. The standard deviation of those scale factors with respect to s_{opt} is computed and if it is smaller than a user-defined threshold, $(\mathbf{X}, \mathbf{Y})$ is considered as a *robust* match and is added to the final list of correspondences $\mathcal{C}$. The robustness of the match stems from the fact that it verifies the robustly extracted scale factor s_{opt} with respect to most (or all) images $I_n \in \mathbf{I}'$. The pairs of center of projections $(\mathbf{C_n^{sfm}}, \mathbf{C_n^{rng}})$ of images in $\mathbf{I}'$ are also added to $\mathcal{C}$.

The list $\mathcal{C}$ contains robust 3D point correspondences that are used for the accurate computation of the similarity transformation (scale factor s, rotation R, and translation T) between the models M_{range} and M_{sfm}. The following weighted error function is minimized with respect to sR and T:

$$E = \sum_{(\mathbf{X},\mathbf{Y}) \in \mathcal{C}} w \|sR \cdot \mathbf{Y} + T - \mathbf{X}\|^2$$

where the weight $w = 1$ for all $(\mathbf{X}, \mathbf{Y}) \in \mathcal{C}$ that are not the centers of projection of the cameras, and $w > 1$ (user-

defined) when $(\mathbf{X}, \mathbf{Y}) = (\mathbf{C_n^{sfm}}, \mathbf{C_n^{rng}})$. By associating higher weights to the centers we exploit the fact that we are confident in the original pose produced by SFM and 2D-to-3D registration. The unknown sR and T are estimated by computing the least square solution from this error function. Note that s can be easily extracted from sR since the determinant of R is 1.

In summary, by utilizing the invariance of the scale factor between corresponding points in M_{range} and M_{sfm}, a set of robust 3D point correspondences $\mathcal{C}$ is computed. These 3D point correspondences are then used for an optimal calculation of the similarity transformation between the two point clouds. This provides a very accurate texture mapping result of the high resolution images onto the dense range model M_{range}.

7. Results & Conclusions

We tested our algorithms using range scans and 2D images acquired from a large-scale urban structure (Shepard Hall/CCNY) and from an interior scene (Great Hall/CCNY). 22 range scans of the exterior of Shepard Hall were automatically registered (Fig. 1) to produce a dense model M_{range}. In one experiment, ten images where gathered under the same lighting conditions. All ten of them were independently registered (2D-to-3D registration Sec. 4) with the model M_{range}. The registration was optimized with the incorporation of the SFM model (Sec. 5) and the final optimization method (Sec. 6). In a second experiment, 22 images of Shepard Hall that covered a wider area were acquired. Although the automated 2D-to-3D registration method was applied to all the images, only five of them were manually selected for the final transformation (Sec. 6) on the basis of visual accuracy. For some of the 22 images the automated 2D-to-3D method could not be applied due to lack of linear features. However, all 22 images where optimally registered using our novel registration method (Sec. 6) after the SFM computation (Sec. 5). Fig. 1 shows the alignment of the range and SFM models achieved through the use of the 2D images. In Fig. 3(a) the accuracy of the texture mapping method is visible. Fig. 3(b) displays a similar result of an interior 3D scene. Table 1 provides some quantitative results of our experiments. Notice the density of the range models versus the sparsity of the SFM models. Also notice the number of robust matches in $\mathcal{C}$ (Sec. 6) with respect to the possible number of matches (i.e., number of points in SFM). The final row Table 1 displays the elapsed time for the final optimization on a Dell PC running Linux on an Intel Xeon-2GHz, 2GB-RAM machine.

We have presented a system that integrates multiview geometry and automated 3D registration techniques for texture mapping high resolution 2D images onto dense 3D range data. The benefits of multiview geometry (SFM) and automated 2D-to-3D registration are merged for the production of photorealistic models with minimal human interaction. Our approach provides a solution of increased robustness, efficiency and generality with respect to previous methods.

	Shepard Hall		Great Hall
Number of points (M_{range})	12,483,568		13,234,532
Number of points (M_{sfm})	2,034	45,392	1,655
2D-images used	10	22	7
2D-to-3D registrations (Sec. 4)	10	5	3
No. of matches in $\mathcal{C}$ (Sec. 6)	258	1632	156
Final optimization (Sec. 6)	8.65 s	19.20 s	3.18 s

Table 1. Quantitative results.

References

[1] Visual Information Technology Group, Canada, 2005. http://iit-iti.nrc-cnrc.gc.ca/about-sujet/vit-tiv_e.html.

[2] P. A. Beardsley, A. P. Zisserman, and D. W. Murray. Sequential updating of projective and affine structure from motion. *International Journal of Computer Vision*, 23(3):235–259, 1997.

[3] P. Besl and N. McKay. A method for registration of 3D shapes. *IEEE Trans. Patt. Anal. and Machine Intell.*, 14(2), 1992.

[4] T. Cass. Polynomial–time geometric matching for object recognition. *IJCV*, 21(1–2):37–61, 1997.

[5] C. Chen and I. Stamos. Semi-automatic range to range registration: A feature-based method. In *The 5th International Conference on 3-D Digital Imaging and Modeling*, pages 254–261, Ottawa, June 2005.

[6] S. Christy and R. Horaud. Iterative pose computation from line correspondences. *CVIU*, 73(1):137–144, January 1999.

[7] O. Faugeras, Q. T. Luong, and T. Papadopoulos. *The Geometry of Multiple Images.* MIT Press, 2001.

[8] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381–395, June 1981.

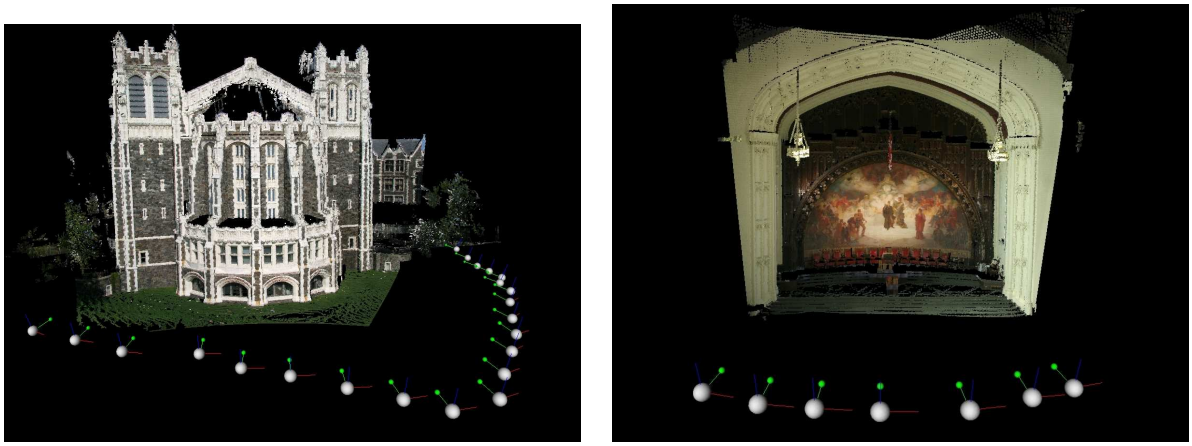
[9] C. Früh and A. Zakhor. Constructing 3D city models by merging aerial and ground views. *CGA*, 23(6):52–11, 2003.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision, second edition*. Cambridge University Press, 2003.

[11] G. Hausler and D. Ritter. Feature–based object recognition and localization in 3D-space, using a single video image. *CVIU*, 73(1):64–81, 1999.

[12] A. Heyden and K. Astrom. Euclidean reconstruction from constant intrinsic parameters. *in Proc. ICPR'92*, pages 339–343, 1996.

[13] R. Horaud, F. Dornaika, B. Lamiroy, and S. Christy. Object pose: The link between weak perspective, paraperspective, and full perspective. *IJCV*, 22(2), 1997.



(a) (b)

Figure 3. **(a)** Range model of Shepard Hall (CCNY) with 22 automatically texture mapped high resolution images. **(b)** Range model of interior scene (Great Hall) with seven automatically texture mapped images. The locations of the recovered camera positions are shown. Notice the accuracy of the photorealistic result.

[14] D. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *IJCV*, 5(7):195–212, 1990.

[15] K. Ikeuchi. The great buddha project. In *IEEE ISMAR03*, Tokyo, Japan, November 2003.

[16] D. W. Jacobs. Matching 3–D models to 2–D images. *IJCV*, 21(1–2):123–153, 1997.

[17] F. Jurie. Solution of the simultaneous pose and correspondence problem using gaussian error model. *CVIU*, 73(3):357–373, March 1999.

[18] Leica Geosystems. http://hds.leica-geosystems.com/.

[19] L. Liu and I. Stamos. Automatic 3D to 2D registration for the photorealistic rendering of urban scenes. In *CVPR*, volume II, pages 137–143, San Diego, CA, June 2005.

[20] D. Lowe. Distinctive image features from scale-invariant keypoints. *Intl. Journal of Computer Vision*, 60(2), 2004.

[21] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models.* Springer-Verlag, 2003.

[22] D. Oberkampf, D. DeMenthon, and L. Davis. Iterative pose estimation using coplanar feature points. *CVGIP*, 63(3), May 1996.

[23] M. Pollefeys and L. V. Gool. A stratifed approach to metric self-calibration. *in Proc. CVPR'97*, pages 407–412, 1997.

[24] M. Pollefeys, L. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, 2004.

[25] K. Pulli, H. Abi-Rached, T. Duchamp, L. G. Shapiro, and W. Stuetzle. Acquisition and visualization of colored 3–D objects. In *ICPR*, Australia, 1998.

[26] L. Quan and Z. Lan. Linear N–point camera pose determination. *PAMI*, 21(7), July 1999.

[27] F. Schaffalitzky and A. Zisserman. Viewpoint invariant texture matching and wide baseline stereo. *Proc. ICCV*, pages 636–643, July 2001.

[28] V. Sequeira and J. Concalves. 3D reality modeling: Photo-realistic 3D models of real world scenes. In *3DPVT*, pages 776–783, 2002.

[29] I. Stamos and P. K. Allen. Automatic registration of 3-D with 2-D imagery in urban environments. *ICCV*, pages 731–736, 2001.

[30] I. Stamos and P. K. Allen. Geometry and texture recovery of scenes of large scale. *Comput. Vis. Image Underst.*, 88(2):94–118, 2002.

[31] I. Stamos and M. Leordeanu. Automated feature-based range registration of urban scenes of large scale. *CVPR*, 2:555–561, 2003.

[32] B. Triggs. Factorization methods for projective structure and motion. *IEEE CVPR96*, pages 845–851, 1996.

[33] A. Troccoli and P. K. Allen. A shadow based method for image to model registration. In *2nd IEEE Workshop on Video and Image Registration*, July 2004.

[34] T. Tuytelaars and L. J. V. Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, 2004.

[35] W. Wells. Statistical approaches to feature–based object recognition. *IJCV*, 21(1–2):63–98, 1997.

[36] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[37] H. Zhao and R. Shibasaki. Reconstructing a textured CAD model of an urban environment using vehicle-borne laser range scanners and line cameras. *MVA*, 14(1):35–41, 2003.

[38] W. Zhao, D. Nister, and S. Hsu. Alignment of continuous video onto 3D point clouds. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1305–1318, 2005.