

Semi-automatic range to range registration: a feature-based method

Chen Chao

Graduate Center of CUNY, New York, NY, CChen@gc.cuny.edu

Ioannis Stamos

Hunter College and Graduate Center of CUNY, New York, NY, istamos@hunter.cuny.edu

Abstract

Our goal is the production of highly accurate photo-realistic descriptions of the 3D world with a minimum of human interaction and increased computational efficiency. Our input is a large number of unregistered 3D and 2D photographs of an urban site. The generated 3D representations, after automated registration, are useful for urban planning, historical preservation, or virtual reality (entertainment) applications. A major bottleneck in the process of 3D scene acquisition is the automated registration of a large number of geometrically complex 3D range scans in a common frame of reference. We have developed novel methods for the accurate and efficient registration of a large number of 3D range scans. The methods utilize range segmentation and feature extraction algorithms. We have also developed a context-sensitive user interface to overcome problems emerging from scene symmetry.

1. Introduction

A typical 3D modeling system involves the phases of 1) Individual range image acquisition from different viewpoints, 2) Noise removal and hole filling for each range image, 3) Segmentation of each range image (i.e. extraction of lines, planes, etc.), 4) Registration of all images into a common frame of reference, 5) Transformation of each range image into an intermediate surface-based or volumetric-based representation, 6) Merging of all range images into a common representation (3D model), 7) Simplification of the final 3D model, and 8) Construction of CAD model of the scene. This paper deals with the semi-automatic registration (fourth task) of a large number of complex 3D scans in the same frame of reference. We present a new and more efficient range-range registration method that complements our original work of [14]. Both of our algorithms (previous and current) are based on the automated feature-matching of lines that have been extracted from a range segmentation

module. This matching leads to coarse pairwise alignment between the scans that is optimized by an Iterative Closest Point (ICP) [3] procedure. The algorithms run under a context-sensitive user interface that can be utilized in cases of incorrect results due to scene symmetry. Our comprehensive solution allows for mm-accurate registration of large scale urban scenes. We present experiments from the registration of three large urban structures.

Most methods that attack the range to range registration problem utilize one of the many variations of the Iterative Closest Point algorithm [3, 15, 2, 11]. In ICP the rigid transformation between two views is iteratively refined, while larger sets of corresponding points between views can be extracted after each refinement step. All ICP-type methods require the meshes to be spatially close with respect to each other in order for an initial set of closest point correspondence to be established. Global ICP-type methods that compute registrations between all acquired scans include the work of Pulli [10] and Nishino [9]. Recently a non-rigid ICP method [4] has been proposed. Hebert [6] introduced the idea of spin-images, where the initial list of corresponding points is extracted by using a pose-invariant representation for the range images. In the approach of [8] a number of roughly pre-registered images are brought into fine alignment by the utilization of a signed distance function that requires sampling of the 3D space. We believe that our method is more efficient for large-scale data sets due to the data reduction induced by our segmentation module. Also, our method does not assume rough pre-registration of the input data sets. On the other hand the user should specify whether two scans overlap or not.

The features used for registration are 3D lines extracted at the borders of segmented planar areas, and at the intersections between segmented planar areas. A solution to the registration problem is possible if **two pairs** of correctly matched lines are found between the two scans S_1 and S_2 . Only the orientation and position of the lines are used due to the fact the endpoints can never be exactly localized (this is an inherent problem of all line detectors). Using these two matched pairs, a closed-form formula provides the de-

sired transformation (R, T) [5, 12]. That means that a blind *hypothesis-and-test* approach would have to consider all possible $\binom{N}{2} \times \binom{M}{2} = O(M^2N^2)$ pairs of lines, where N and M are the number of lines from scans S_1 and S_2 respectively. Such an approach is impractical due to the size of the search space to be explored. For each pair of lines we would need to compute the transformation (R, T) and then verify the transformation by transforming all lines from scan S_1 to the coordinate system of scan S_2 . The algorithm [14], previously developed by our group, provides a solution to the problem of this large exploration space, by rejecting a large number of line pairs before the expensive verification step. In our previous work, the additional information of the 3D plane on which each extracted line lies facilitates the search. Also, the length of the lines, and the size of the planes is used in order to discard invalid pairs at a preprocessing step, and in order to verify the quality of the match at later steps. The central idea is the selection of a robust pair of lines for the computation of an exact rotation and of an estimated translation and of a second pair of lines in order to evaluate the computed transformation.

In this paper a new complementary and more efficient algorithm is introduced. This algorithm is supported by a context-sensitive user interface. Our registration system first extracts three major directions from each range image by applying a range segmentation step [13] and by clustering the extracted linear segments and plane normals. A local object-based coordinate system for each range image is constructed next, by computing three major orthogonal axes. The rotational transformation between pairs of scans can be computed quickly and accurately by matching these major scene axes between the scans. Candidate translations are then estimated by matching linear segments between pairs of rotationally aligned range images. Finally, these candidate translations are clustered using an unsupervised nearest-neighbor classification method. The correct translation vector should be in one of the major clusters of translations (i.e. being the one appearing most frequently). This maximization criterion though can lead to wrong registration when the 3D scene appears symmetric either rotationally or translationally. A user interface (see Sec. 3) has been designed to deal with the previously mentioned cases. The user interface has the following characteristics: a) It prompts a user to indicate whether the registration is correct or wrong; b) It displays other possible rotations between pairs if the automatically computed one is incorrect; and c) It allows for the direct adjustment on the transformation by rotating one image scan and moving it along the three major scene directions.

2. Automated Registration Method

All overlapping pairs of range scans are considered and the transformation between the two scans is computed. Our new automated registration process involves three steps: 1) line and plane clustering, 2) rotation estimation, and 3) translation estimation. When the transformations between all pairs are computed and verified by the user, an ICP routine optimizes the pairwise transformations. Finally, a global registration procedure computes the transformation of all scans with respect to a selected pivot scan to stitch all scans into a common coordinate system. Note that this procedure does not optimize the pairwise registrations already computed. Fig. 1 shows the flowchart of our system.

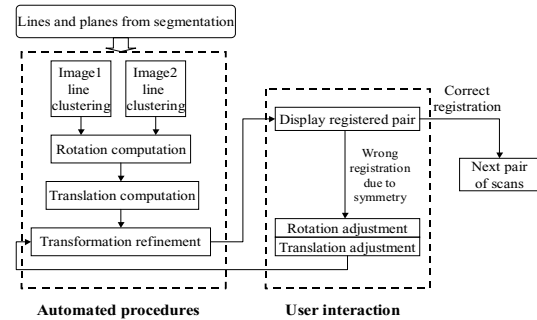


Figure 1. Flowchart of range to range registration with user interface.

2.1. Rotation Estimation

Man-made urban scenes are characterized by sets of linear features organized in a major vertical and a number of horizontal directions. After the segmentation phase [13], the extracted 3D line directions and plane normals are clustered into three major 3D directions (Fig. 2). The clustering procedure groups all the line vectors into clusters (a vector becomes part of the cluster if its angle from the centroid of the cluster is smaller than angle threshold). In most cases this procedure extracts three major directions that are perpendicular to each other. In the cases that only two major clusters are found, we can obtain the third major direction as the cross product of the extracted two. Thus, our main assumption is that our 3D scene contains at least two major perpendicular directions. This is a reasonable assumption that is commonly used in urban scene settings (see [1]).

After obtaining three axes $\{X_1, Y_1, Z_1\}$ from the left image, and $\{X_2, Y_2, Z_2\}$ from the right image, all possible values for the rotational matrix R that rotates (X_1, Y_1, Z_1) to (X, Y, Z) are computed. (X, Y, Z) is any permutation of $(\pm X_2, \pm Y_2, \pm Z_2)$. There are 24 such rotation matrices that rotate the left image into the coordinate system of the right

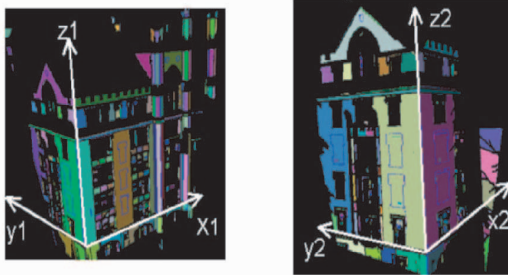


Figure 2. Three major scene directions extracted from two segmented range scans (different colors correspond to different segmented planes). A correct match between the directions provides a unique solution for the rotational transformation between the scans.

one. However using simple heuristics, the number of candidate rotations can be significantly reduced.

The position of each 3D point recorded refers to the range scanner's inner coordinate system, which is defined as shown in Fig. 3. The laser generator/receptor is the origin point of the local coordinate system. The negative Z axis points towards the 3D scene. Let us consider how the coordinate system changes from one scan to another. In our algorithm, we always choose the right image as the pivot image, and transforms all the point coordinates in the left image into it. If the rotation matrix is:

$$R = \begin{bmatrix} R_{00} & R_{01} & R_{02} \\ R_{10} & R_{11} & R_{12} \\ R_{20} & R_{21} & R_{22} \end{bmatrix},$$

then the unit vector $[0, 1, 0]^T$ representing Y_1 axis would be transformed into a unit-vector $[R_{10}, R_{11}, R_{12}]^T$ in the right image. In other words, R_{10} , R_{11} and R_{12} are the projections of Y_1 onto the X , Y and Z axes. During the scanning process, the scanner is mostly moving on the ground level, with at most 45° tilt-angle of the Y -axis. Since the Y -axis does not change dramatically, we select a threshold for R_{11} that is at least 0.7. Similarly, successive images are likely to be from close-by viewpoints, so the R_{00} and R_{22} are restricted to have positive values. With these restrictions, the candidates for rotation becomes fewer than 5, and in some cases, 2 or 3. Then with the assumption that successive images are close to each other, we order these candidate rotation matrices by the sum of the diagonal elements, $R_{00} + R_{11} + R_{22}$, and choose the one with the largest sum as the rotation matrix. The other possible rotations are kept as candidates. Note that these assumptions can be relaxed without affecting the actual outcome, since we can choose from the candidate rotations the correct one. The described heuristic is used in order to speed up our algorithm.

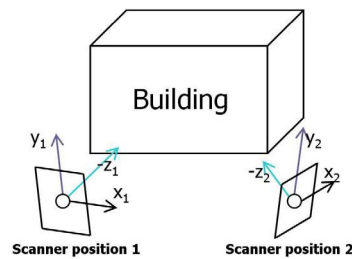


Figure 3. Range scanner's local coordinate systems at two different viewpoints.

2.2. Translation Estimation

Once the rotation has been automatically computed (see Sec. 2.1), or manually selected (see Sec. 3), the axes of the two local coordinate systems of the two scans can be aligned. The next step is the estimation of the translational vector between the two range images. We search for matching pairs of 3D linear segments between the two scans, since two correctly matched pairs provide a unique solution for the translation. At a preprocessing stage, the distance between each pair of parallel linear segments is computed. This distance is the vector that connects the midpoints of the two segments (Fig. 4). From every two lines in the left image ($l1, l2$) and two lines in the right image ($r1, r2$), a candidate translation is computed if and only if:

1. All four lines are parallel to each other, and the distance between $l1$ and $r1$ equals (within a length and angle threshold) the distance between $l2$ and $r2$ (Fig. 4(a)). In this case, the average of the two distances is recorded as a candidate translation.
2. Lines $l1, r1$ are parallel to each other, and lines $l2, r2$ are parallel to each other, but lines $l1, l2$ are not parallel. In addition the distance between $l1$ and $r1$ is equals (within a length and angle threshold) the distance between $l2$ and $r2$ (Fig. 4(b)). In this case, an exact translation can be computed by the solution of an over-constraint linear system as explained in [12].

The computed candidate translations are then clustered into groups of translations that are close to each other within certain thresholds of length and direction. Intuitively, the correct translation is the one that occurs most frequently. This is the one that defines the largest cluster of candidate translation. However in order to take into account measurement noise and scene symmetry, we consider the N ($N=10$) largest clusters of candidate translations. The centroids of these N clusters are considered as our final candidate translations. Finally, out of these N centroids the one that maximizes the number of line matches between the two scans is

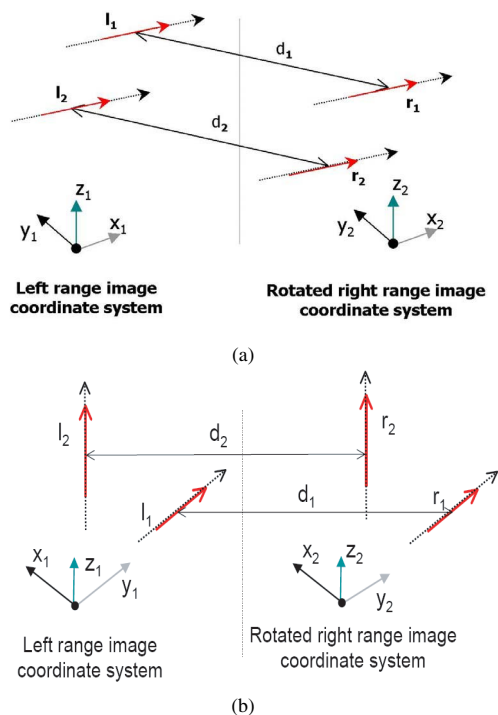


Figure 4. Two pairs of matched lines. The vectors connecting the midpoints of the matched segments provide approximations to the translation between the two scans. (a) All four lines parallel to a same axis. (b) Two line pairs parallel to different axes.

returned as the final translation vector¹.

The above automated procedure computes a transformation between any pair of images that overlap. The registered image pair is then displayed in the user interface (Sec. 3). The registration obtained after this stage is very accurate, but still not quite satisfying for photorealistic 3D modeling. The inaccuracy comes from several factors: a) The segmentation phase introduces some errors in extracted line directions and lengths, and b) The clustering methods for rotation estimation and translation estimation introduce errors as well. In the clustering of 3D lines the centroid of each cluster is selected as the representative major direction. Also in the clustering of candidate translations, the centroid of each cluster is selected as the representative translation. That is why, in order to minimize the registration error an ICP algorithm needs to be applied as a post-processing step. Given that the registration from the automated routine

¹The number of lines that match assuming a rotational matrix and translational vector can be computed after both scans are placed on the same coordinate system. See [14].

and user interaction is very close to the exact registration, the ICP algorithm is then able to optimize the overlapping points of two image scans. In Sec. 4 experimental results show that the registration error is greatly decreased after ICP optimization and it reaches the level of a few mm.

3. Context-Sensitive User Interface

In order to visualize the procedure of registration, as well as to allow users to correct wrong registrations due to 3D scene symmetry, a context-sensitive user interface has been developed. For each pair of overlapping scans the system reads the segmented planar areas and linear segments. The efficient range-range registration algorithm described in the previous sections is being executed first. After a few seconds the result is displayed (Fig. 5). If the user is satisfied with the result s/he can proceed with the next pair of scans. If on the other hand there is a mistake the system displays the following options:

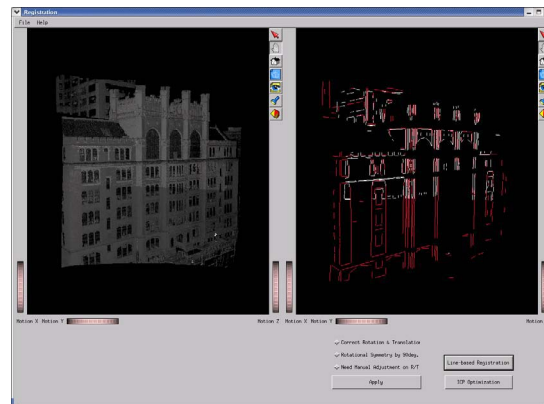


Figure 5. Overview of the user interface. Two automatically aligned range scans are shown. Left window: raw range scans. Right window: the same scans abstracted as linear segments (different colors are used for different scans).

(1) If the initial rotational calculation was wrong due to an erroneous match of axes the user is presented with a set of possible orientations (Fig. 6). The user can select the correct orientation. The system then recalculates the translation (see Sec. 2.2), and s/he is asked to verify the result. (2) If the rotational calculation was correct but the result is still wrong, then the user may choose to invoke the more expensive and complementary range-range registration algorithm described in [14]. (3) If no automated algorithm provides a correct result then the user needs to manually fix the resulted transformation. Note that this case can appear due to symmetry of the acquired 3D scene. Fig. 7 shows the screen

that the user sees. (4) After the user manually corrects the transformation the refinement procedure that searches for matching features between all lines can be invoked.

We call this user-interface context-sensitive because the user can translate or rotate the 3D scans only among the major axes that form the object's local coordinate system as shown in Fig. 7. The three axes of the right image are displayed as red lines, along each of them there is a translation dragger, and a rotation ball. By dragging each dragger, the translation on one direction is adjusted independently, and thus overlapping lines and points can be easily adjusted to the best accuracy; the rotation ball is used to adjust rotation around each axis: by dragging the ball along that axis, its translation is transformed into a rotation around that axis by the corresponding angle, which is applied to the left image. By alternatively adjusting the rotation and translation, the manual registration becomes a lot easier and more accurate than other methods of alignment such as picking three corresponding points from both images, or translating the two scans along axes that are not related to the geometry of the scene.

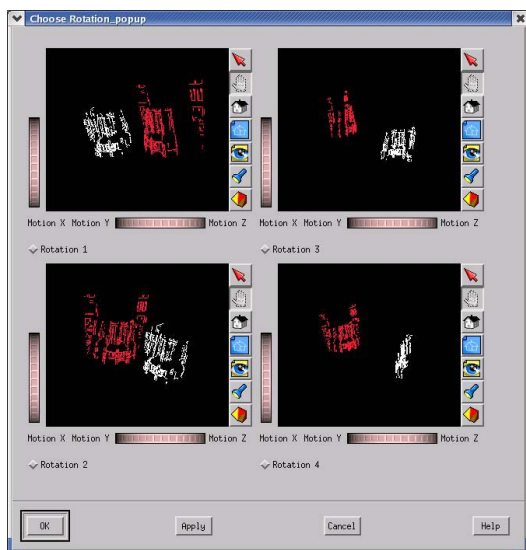


Figure 6. A set of possible orientations between the two coordinate systems is presented to the user to choose from. In this example the rotation in the upper left corner corresponds to the correct result.

4. Experiments and Conclusions

We tested the semi-automatic registration system on two urban structures of different styles. The Thomas Hunter building (Hunter College of CUNY) is a rectangular build-

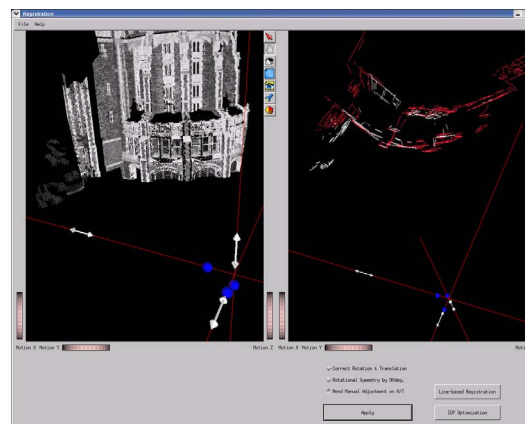


Figure 7. The user can manually translate or rotate one scan with respect to the other. This task is made much simpler due to the fact that the user can translate along or rotate about the major orientations of the 3D scene.

ing with flat side walls. The Shepard Hall building (City College of CUNY) has a more complicated architecture that resembles a Gothic cathedral. We also tested our algorithm using scans gathered from the interior of the Shepard Hall building. Laser range scans were acquired by a Cyrax 2500 laser range scanner. Each scan consists of a million points with an accuracy of 6mm per point. As a criterion of registration performance, we record the number of matching line pairs as computed in Sec. 2.2 (Fig. 8), and we calculate the average distance between matching planes.

Considering the Thomas Hunter building data, we registered 14 range images by applying 15 pair-wise registrations. Among these pairs, 13 pairs were correctly registered with the automated routine and refined by the ICP optimization. Two pairs require the user to adjust the translation and rotation before a correct registration was obtained. The time for each automated registration is displayed in Table 1 (top) (on average 20 seconds per pair - 2GHz Xeon Processor - 2Gbit RAM). Table 1 (top) also shows the average distance between matched planes² of registered pairs of scans, as well as how much the ICP optimization further improved the accuracy of registration. The average error over all pairs of scans decreases from 21.17 mm (before ICP) to 1.77 mm (after ICP). The final registered line and point images are shown in Figs. 10(a), and 10(b).

Table 1 (middle) shows the pair-wise registration time and error measurements for Shepard Hall (15 pairs shown). Since this building has more delicate geometric features, the segmentation produces a large amount of short line segments in various directions. Nevertheless the experimental

²Each extracted 3D line lies on the border of a segmented planar region. Therefore matched lines between scans dictate matched planar regions.

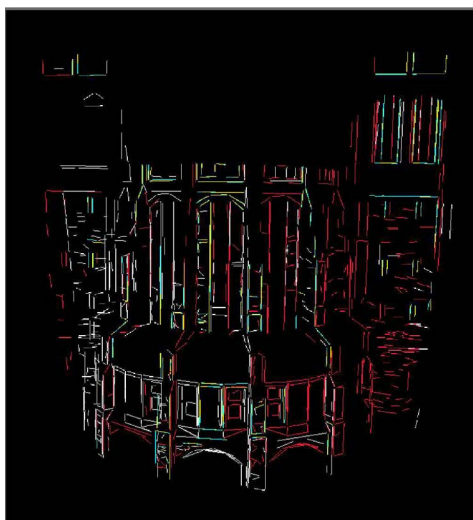


Figure 8. Matching lines between two scans. White/red lines are border lines, and yellow/blue lines are the matching lines from two images respectively.

results show that the algorithm is quite robust: among the 24 pairs of scans, 9 pairs were automatically registered, 8 pairs needed manual translational adjustment due to scene symmetry, and 7 pairs required a careful user adjustment on rotation. Because of this, the total time of the registration is about an hour (this includes user interaction), although the automated registration on each pair takes less than one minute. When the rotation needs to be manually adjusted, the resulted registration usually has quite visible registration errors, as shown in Fig. 9(a). In this case, ICP optimization greatly improves registration accuracy (Fig. 9(b)). The final registered line and point images of Shepard Hall are shown in Figs. 10(c) and 10(d). The average error over all pairs of scans decreases from 51.72 mm (before ICP) to 3.23 mm (after ICP).

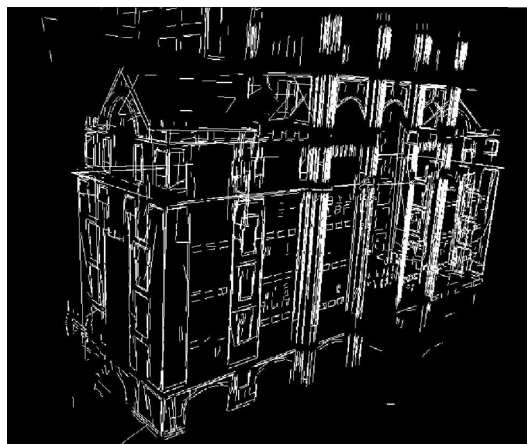
We also performed experiments in the interior of the Shepard Hall and registered 21 scans (Figs. 10(e) and 10(f)). Out of 44 pairs the automated procedure produced 12 correct results, whereas 18 results needed translational adjustment due to scene symmetry and 13 need manual adjustment of translation and rotation. The average error over all pairs of scans improves from 17.59 mm (before ICP) to 7.26 mm (after ICP) (Table 1 (bottom)). Note that in most cases the number of matching line pairs increase after the ICP optimization (this is what is expected when the scans are brought closer to each other). In some cases though the number of matching lines decreases, without the registration quality to be sacrificed. On the contrary we can see

from the average plane error that ICP further improved the registration accuracy.

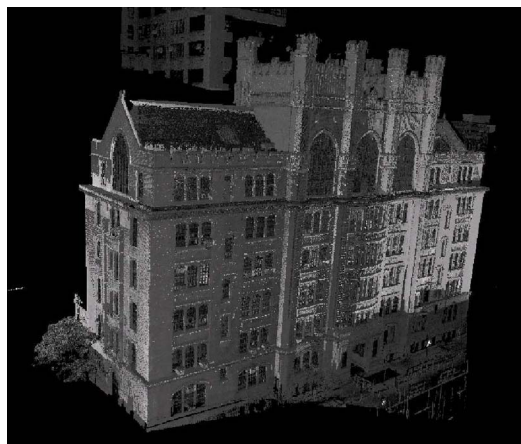
We have presented a semi-automatic registration system that incorporates an automated range-range registration algorithm with a context-sensitive user interface. The user interface is being utilized in all cases of registration errors produced by scene symmetry. This system complements our original work of [14] and produces efficiently high-quality registration results. We believe that we have built an arsenal of methods that can be utilized for the automatic registration of large-scale urban scenes. Our future work includes the development of a method for global optimization after all pairwise registrations have been achieved. This will significantly improve the final result. We are also working on automated registration in scenes that do not contain a plethora of linear features. Finally, we have utilized matching algorithms between lines for the solutions of 3D range to 2D image registration in urban scenes [7].



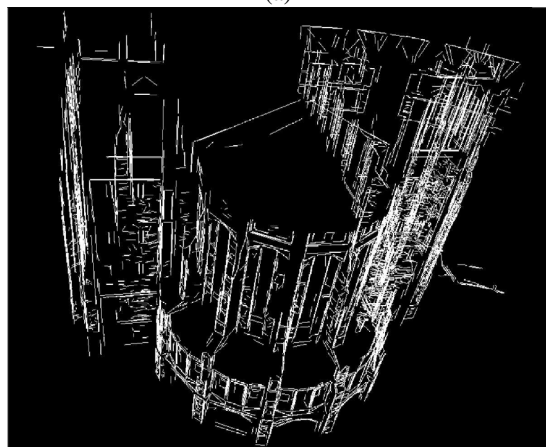
Figure 9. Shepard Hall. Close up view of pairwise registration. (Top) With automated registration before ICP optimization. Range scans do not align perfectly. (Bottom) After ICP optimization. Result has been significantly improved.



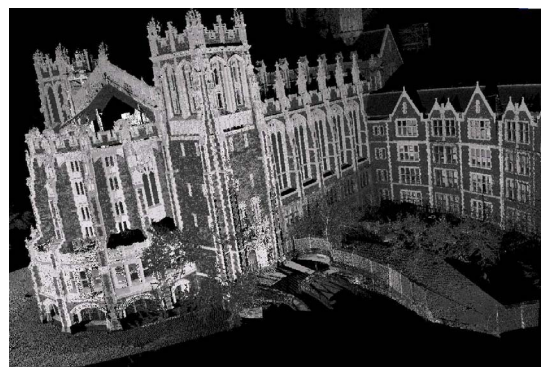
(a)



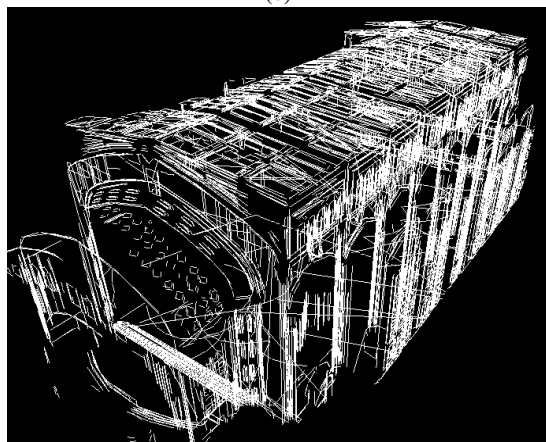
(b)



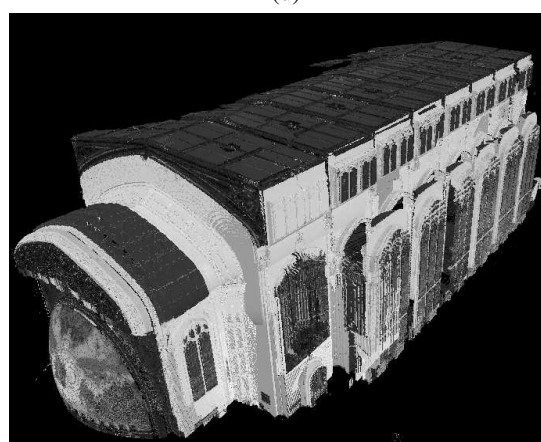
(c)



(d)



(e)



(f)

Figure 10. Registration results. (a) & (b) Thomas Hunter building (14 scans). (c) & (d) Shepard Hall building (24 scans). (e) & (f) Interior of Shepard Hall (21 scans). Registered line and range images shown. The lines are extracted from the range segmentation module. The range images correspond to the source scans. The gray values correspond to the returned laser intensity.

Pair	Line Pairs	t	Before ICP		After ICP	
			N	P_{err}	N	P_{err}
1	322x229	19	10	33.95	26	1.80
2	322x275	19	19	5.87	17	1.68
3	243x205	2	7	54.70	11	1.72
4	205x292	6	6	5.15	7	0.97
5	292x279	38	12	15.12	36	1.87
6	279x275	20	21	7.72	20	0.91
7	275x304	31	50	14.09	32	1.03
8	304x180	23	22	22.51	22	2.98
9	195x180	32	19	3.85	33	1.02
10	195x249	28	12	15.74	27	2.04
11	180x249	4	6	50.74	18	1.60
12	129x249	31	13	5.66	31	2.50
13	249x137	19	6	24.79	26	3.16
14	129x137	29	7	19.32	37	2.11
15	137x332	9	7	38.36	9	1.23

Pair	Line Pairs	t	Before ICP		After ICP	
			N	P_{err}	N	P_{err}
1	625x211	21	3	52.64	8	11.94
2	546x539	43	34	78.05	88	1.80
3	546x638	56	8	42.60	9	3.20
4	546x211	31	3	97.26	42	2.64
5	539x638	45	27	85.71	31	3.51
6	638x642	62	113	4.78	112	1.95
7	638x360	17	30	57.39	28	2.42
8	642x360	28	17	9.49	16	2.81
9	708x237	8	8	16.93	8	3.79
10	734x334	14	12	83.59	8	0.52
11	334x149	6	4	47.02	18	1.71
12	149x176	3	7	51.48	37	1.18
13	649x501	33	23	21.33	21	3.28
14	501x203	10	24	9.59	24	5.05
15	203x281	4	8	117.90	11	2.63

Pair	Line Pairs	t	Before ICP		After ICP	
			N	P_{err}	N	P_{err}
1	787x645	36	147	9.71	138	1.61
2	654x787	21	41	16.34	25	2.63
3	654x638	24	252	13.31	124	3.28
4	356x351	13	84	8.12	68	1.74
5	174x283	2	42	13.90	36	5.62
6	585x557	28	56	26.33	137	14.73
7	656x606	45	249	10.24	138	11.65
8	656x654	41	257	11.03	160	19.62
9	656x481	4	13	31.14	19	3.72
10	654x585	16	7	40.12	11	1.19
11	654x910	33	121	6.11	118	0.99
12	910x864	44	268	14.53	128	2.00
13	647x787	43	84	6.34	89	1.86
14	647x356	13	5	49.04	17	37.61
15	647x619	8	51	7.61	36	0.63

Table 1. Experimental results on Thomas-hunter building (top), Shepard Hall exterior (middle) and interior (bottom). t : time of automated registration (before ICP optimization) in secs ; N : number of matching lines between the two scans; P_{err} : average distance between matching segmented planar regions (in mm).

5. Acknowledgments

Supported in part by NSF CAREER IIS-01-21239, NSF MRI/RUI EIA-0215962. We would like to thank Prof. Wolberg and his group (City College of CUNY) for the acquisition of the City College range scans.

References

- [1] M. E. Antone and S. Teller. Scalable extrinsic calibration of omni-directional image networks. *IJCV*, 49(2/3):143–174, Sept./Oct. 2002.
- [2] F. Bernardini, H. Rushmeier, I. M. Martin, J. Mittleman, and G. Taubin. Building a digital model of Michelangelo’s Florentine Pietà. *IEEE Computer Graphics and Applications*, 22(1):59–67, /2002.
- [3] P. J. Besl and N. D. Mckay. A method for registration of 3–D shapes. *IEEE Trans. on PAMI*, 14(2), Feb. 1992.
- [4] B. Brown and S. Rusinkiewicz. Non-rigid range-scan alignment using thin-plate splines. In *Symposium on 3D Data Processing, Visualization, and Transmission*, Sept. 2004.
- [5] O. Faugeras. *Three-Dimensional Computer Vision*. The MIT Press, 1996.
- [6] D. Huber and M. Hebert. Fully automatic registration of multiple 3D data sets. In *IEEE Comp. Soc. Workshop on Comp. Vis. Beyond the Visible Spectrum*, Dec. 2001.
- [7] L. Liu and I. Stamos. Automatic 3D to 2D registration for the photorealistic rendering of urban scenes. *CVPR*, San Diego, 2005 (to appear).
- [8] T. Masuda. Registration and integration of multiple range images by matching signed distance fields for object shape modeling. *CVIU*, 87:51–65, July 2002.
- [9] K. Nishino and K. Ikeuchi. Robust simultaneous registration of multiple range images. In *ACCV2002: The 5th Asian Conference on Computer Vision*, Jan. 2002.
- [10] K. Pulli. Multiview registration for large datasets. In *Second Int. Conf. on 3D Dig. Im. and Modeling*, 1999.
- [11] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-time 3D model acquisition. In *SIGGRAPH*, 2002.
- [12] I. Stamos. *Geometry and Texture Recovery of Scenes of Large Scale*. PhD thesis, Columbia University, 2001.
- [13] I. Stamos and P. K. Allen. Geometry and texture recovery of scenes of large scale. *CVIU*, 88(2):84–118, Nov. 2002.
- [14] I. Stamos and M. Leordeanu. Automated feature-based range registration of urban scenes of large scale. In *CVPR*, volume II, pages 555–561, Madison, WI, June 2003.
- [15] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *SIGGRAPH*, 1994.