Automatic Registration of Large-Scale Multi-sensor Datasets

Quan Wang and Suya You

Computer Science Department, University of Southern California,
Los Angeles, California, U.S.A.
{quanwang,suyay}@graphics.usc.edu

Abstract. This paper proposes an automatic method for registering images from different sensors, particularly 2D optical sensors and 3D range sensors, without any assumption about initial alignment.

Many existing methods try to reconstruct 3D points from 2D image sequences, and then match 3D primitives from both sides. The availability of appropriate multiple images associated with 3D range data, the well-known challenge of inferring 3D from 2D and the difficulty of establishing correspondences among 3D primitives when there is no pre-knowledge about initial pose estimation, lead us to a different approach based on region matching between optical images and depth images projected from range data.

This paper details our interest region extraction method for optical images and also the efficient region matching component. Experiments using several cities' aerial images and LiDAR (Light Detection and Ranging) data illustrate the effectiveness of the proposed approach even when facing considerably geometric distortions.

Keywords: different sensors registration, 2D-3D matching, LiDAR data.

1 Introduction

Recent years, there has been an increasing awareness of the growing need for registering images from different sensors, especially the range and optical sensors. For example, the photorealistic modeling of urban scenes using range data from airborne or ground laser scanner requires the registration of those 3D range data onto aerial or ground 2D images for recognition and texture mapping purposes. In the medical image processing domain, there has a long standing concern about how to automatically align Computed Tomography or Magnetic Resonance images with optical camera images. Traditional texture-based image matching approaches such as [1] can not be directly adapted to above tasks, basically because unlike the optical sensors, range sensors capture no texture information.

In this paper, we propose an automatic registration method based on matching of local interest regions extracted from 2D images and depth images of 3D range data for urban environment. The regions we are interested in (ROI) are typically well separated regions of individual buildings. Global context information is implicitly

K.N. Kutulakos (Ed.): ECCV 2010 Workshops, Part II, LNCS 6554, pp. 225–238, 2012.

used for outlier removal and matching propagation (system overview in figure 1). Our approach can register images from different sensors with large initial location and scale errors. Although today there exist systematic ways to obtain initially well aligned 3D and 2D data at the same time for large scale scenes, possible applications of our work include data fusion from different sources and sensors, and updating existing GIS (Geographic Information System) with new content, when data from different sources may have non-unified calibration or no georeference at all, e.g. historic photos or photos from common users. Furthermore, the ROI extraction component proposed in this paper is an important prerequisite to a variety of recognition, understanding, and rendering tasks in urban environments.

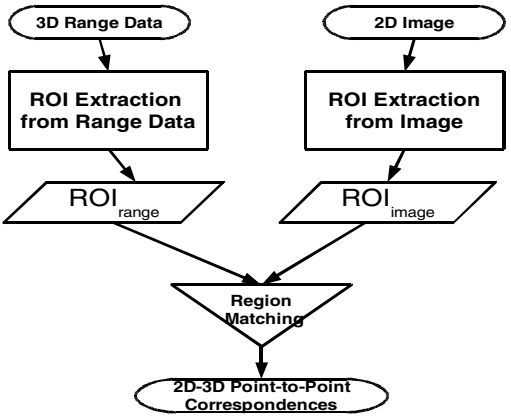

Fig. 1. Overview of the proposed 2D-3D registration system

Our two basic assumptions are: first the dominant contours of most interest regions are repeatable under both optical and range sensors. A similar assumption was used and verified in [8]. Second, focusing on different sensor problems, in this paper we assume both optical and depth images have similar viewing directions (nadir view in our experiments) though position, zoom level and in-plan rotations of capture devices can be different. Our idea for the whole system is to first handle different sensor problems in this stage, and then register nadir, oblique and even ground images all from optical sensors to handle 3D view point changes by using approaches such as [13] and [16]. In the end, oblique and ground images can be indirectly registered.

Intensive experiments have verified the effectiveness of the proposed approach in terms of scale, rotation and location invariance, significant geometric distortion and partially missing data due to occlusion or historic data. After the related works, section 3 details our interest region extraction method for optical images and section 4 presents the region matching component.

2 Related Works

To register images from different sensors, many recently developed methods reconstruct sparse or dense 3D point clouds from image sequence, then use high level

features (e.g. 3D edges, intersection of perpendicular 3D lines) which are preserved and consistent on both 3D and 2D sides to establish correspondences.

Zhao, et al. [2] use motion stereo to recover dense 3D point clouds from continuous oblique video and ICP algorithm to register recovered 3D points with LiDAR data with initial alignment provided by positioning hardware such as GPS (Global Positioning System) and IMU (Inertial Measurement Unit). Ding, et al. detect 2D orthogonal corners (2DOC) and use them as primitives to match single oblique image with airborne LiDAR [3]. The proposed method achieves overall 61% accuracy and the processing time of each image is only several minutes in contrast to 20 hours of the previous work of [4].

Both [2] and [3] utilize positioning hardware for initial alignment. Our own visualization of similar datasets indicates the readings from the airplane-bonded GPS and IMU are accurate enough to significantly simplify the registration problem. However, for historic data or photos from common users, we can not assume such assistant hardware is always available for GIS data fusion and updating problems. Moreover, though accurate for large city scenes, the current accuracy of positioning hardware makes their application to small scenes (e.g. indoor environment and medical imaging settings) impractical. If initial orientation, scale and location errors are significant, ICP or local search of orthogonal corners could not be sufficient.

Multiview geometry methods are used in [2] and [6] to recover 3D point clouds from image sequences. The first limitation is appropriate multiple views of the interest object might not always be readily available. Second, as the first step of 3D reconstruction, correspondence among 2D images needs to be established. This is a challenge problem by its own especially for wide baseline cases. Simple Harris corners and correlation are used in [2] for continuous video frames, while in [6], SIFT is use. However, for non-planar 3D object and significant view point changes, even SIFT and its many variations can not be confidently counted for dense and stable correspondences. Last but not the least, even a number of perfect correspondences can be obtained , traditional stereo or structure from motion techniques still tend to produce inconsistent and noisy results.

3 ROI Extraction from Aerial Images

One important component of our 2D-3D registration method is ROI extraction from aerial images (major components in fig. 2), which can be viewed as a special case of general image segmentation problem. Related recent works include: Comaniciu and Meer's non parametric mean shift segmentation algorithm [9] and Felzenszwalb and Huttenlocher's efficient graph based segmentation methods [10].

The fractal geometry used in our method, originally introduced by Mandelbrot [11], has long been used for aerial image understanding tasks. Solka et al. use fractal measurement combined with classical statistical features such as the coefficient of variation to identify ROI for unmanned aerial vehicle imagery [12]. Recent work of Cao et al. [14] tries to minimize an energy function representing how well the current boundary contains the interest region using fractal error image and texture edge image generated by Discrete Cosine Transformation.

This section presents our ROI extraction algorithm for aerial photography. The proposed algorithm produces initial ROI through a region-growing process utilizing various image cues from low level features such as intensity and color preference to high level ones such as fractal errors and multiple assistant information maps (AIMs). The detected initial ROI could be further refined by a learning-based region regulation step. This component is to extract, from aerial images, buildings' most external contours repeated and consistent with those extracted from 3D range data.

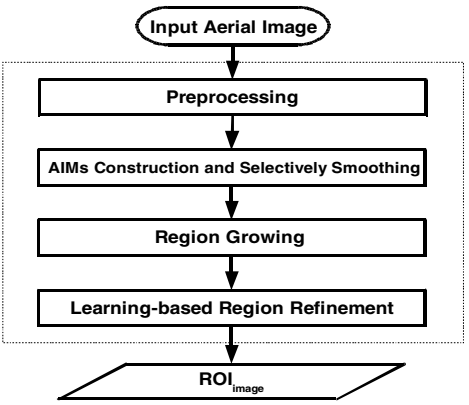

Fig. 2. ROI extraction from aerial images

3.1 AIMs Construction and Selectively Smoothing

There are three kinds of assistant information maps the region growing process frequently refers to: vege maps, shadow maps and edge maps. The aerial image is also selectively smoothed during the construction of three AIMs.

Vege-Map ($\mathbf{M_{vege}}$): By utilizing color information in the aerial photograph, we identify pixels that are dominated by the green channel and possibly vegetations.

Shadow-Map ($\mathbf{M_{shadow}}$): For each pixel, let I represent the intensity value and (C_r, C_g, C_b) represent its RGB color channels. A pixel is said be to a shadow pixel if:

$$I < T_{shadow1} \text{ and } \max\{C_r, C_g, C_b\} < T_{shadow2} \quad (1)$$

where $T_{shadow1}$ and $T_{shadow2}$ are thresholds specifying how low the intensity and color channel need to be for a shadow pixel. Because vegetations typically form low reflection regions, the shadow-map typically have many overlaps with the vege-map.

Edge-Map ($\mathbf{M_{edge}}$): There are two kinds of edges in our edge-map, the true edges and the in-region edges. Among the initial edges returned by Canny operator, most are not actual boundaries of ROI (true edges) but rather edge responses within those regions (in-region edges) due to slope or textures of the roofs, items like air conditioners on the building's top, or even noises from image sensors.

The existence of in-region edges is one primary reason for over-segmentation. Moreover, since our ROI extraction process is a combination of region-driven and edge-driven, it is meaningful to distinguish those two kinds of edges from the very

beginning. For urban scenes with regular buildings, an edge pixel is deemed as a part of true edges unless neighboring horizontal or vertical non-edge pixels have similar hues. HSV instead of RGB color space is used because neighboring pixels of either true or in-region edges tend to be affected by different lighting, and hue is generally more robust under such circumstance. The separation of in-region edges from true edges serves two purposes. First, while the true edges will become strict barrier during the region-growing process, those in-region pixels will not. The region-growing process is allowed to pass those in-region pixels with certain penalty to the confidence attribute. Second, we perform selectively smoothing based on the results of in-region edges. The color and intensity of each confirmed in-region edge pixel will be replaced by the average of its non-edge eight neighbors, helping us eliminate those in-region details which will otherwise compromise segmentation performance.

3.2 Region Growing

A uniform grid is placed on top of aerial images to determine seed locations. Each cell's center P is used as a tentative seed location and if it fails the seed conditions:

$$P \notin M_{vege} \text{ and } P \notin M_{shadow} \text{ and } P \notin M_{edge} \tag{2}$$

the cell is equally divided into four smaller cells and each center of those four sub-cells is tested again. It is possible that all five tests fail and the corresponding cell has no marker at all (e.g., when the cell is placed on trees).

During the region growing process, the current pixel ($p_{current}$) will be accepted and recursively expanded only if it meets the three expansion requirements:

1) The fractal error requirement: The theory is based on the properties of nature features to fit a fractional Brownian motion model. The definition of fractal error in image domain concerns two pixel locations (p_c and p_r). The measurement (e.g. intensity) difference of those two locations should be normally distributed with a mean of zero and a variance proportional to the 2H power of the Euclidean distance.

For intensity measurement, if the model fits, the average absolute intensity change across several pairs of pixels should follow exponential scaling:

$$E[|I(p_c) - I(p_r)|] = k \mid p_c - p_r \mid^H, \tag{3}$$

where E is the topological dimension (the number of independent variables) and in the image domain E = 2. $k > 0$ and $0 < H < 1$ are two parameters. The parameter H is related to the fractal dimension D by: D = E + 1 -H.

The above equation can be linearized by logarithm:

$$\ln(E[I(p_c) - I(p_r)]) = \ln(k) + H \ln(\mid p_c - p_r \mid). \tag{4}$$

With the linear equation, we can use machine learning technique to obtain the estimates of H and k. To obtain training data, a window operator is placed on one aerial image's non-building regions. After collecting pixel distances and their associated intensity changes in those regions, the least-squares linear regression is used to compute the optimized $\overline{H}$ and $\overline{k}$.

The individual fractal error for a pixel location p_c is calculated as the difference between the actual and estimated values from one of its neighboring pixel p_r:

$$F_{error}(p_c, p_r) = E[I(p_c) - I(p_r)] - \bar{k} \mid p_c - p_r \mid^{\overline{H}}. \quad (5)$$

Finally, the overall fractal error (OFE) for p_c is computed as the root mean square (RMS) of these individual errors using a local window centered at p_c:

$$OFE_{p_c} = \sqrt{\frac{1}{n}\sum_{p_r} F_{error}(p_c, p_r)}, \quad (6)$$

where n is number of pixels considered in a local window.

A low OFE indicates that the center pixel's neighboring region is more likely to belong to a non-building region. Therefore, the center pixel will be excluded from the current growing region. A center pixel with sufficient high OFE will pass this expansion requirement. We never compute a fractal map for the entire aerial image because there are many regions in the aerial images that are never reached throughout the region-growing process due to one expansion requirement or another. Instead, we take the compute-on-demand-then-save way.

2) Requirements from AIMs: The current pixel will fail this requirement if $p_{current} \in M_{vege}$ or $p_{current} \in M_{shadow}$. The requirement for shadow-map can be relaxed in heavily urbanized scenes with long shadows overlapping buildings. If the current pixel belongs to an in-region edge, it will still pass this test though a penalty to this region's confidence needs to be taken. If the current pixel belongs to a true edge, it will be neither accepted nor further expanded.

3) The dynamic intensity range: Finally the current pixel's intensity must lie within the current dynamic intensity range, defined by two variables: the upper bound (U_{range}) and the lower bound (L_{range}). Both are initialized as the intensity of initial seed point. The range is expanded simultaneously with the region growing process with a limit for the range's length (*range_len*).

The current pixel will immediately pass the dynamic intensity range requirement without any update if:

$$L_{range} < I(p_{current}) < U_{range} \quad (7)$$

Otherwise, we introduce a tolerate threshold T_{range} as an expansion limit. The threshold is softened and fluctuated based on the current area to handle the case when the current point falls into a small distinct region contained in a large region we are interested in. The current pixel will still pass this requirement and update the range if: when the current area is smaller than the minimum acceptable area,

$$\begin{aligned} I(p_{current}) &> L_{range} - (T_{range} + (-e^{(cur\_area - Area_{\min})} + 1) \cdot Range_{\max}) \\ I(p_{current}) &< U_{range} + (T_{range} + (-e^{(cur\_area - Area_{\min})} + 1) \cdot Range_{\max}) \end{aligned} \quad (8)$$

or when the current area is larger,

$$L_{range} - T_{range} < I(p_{current}) < U_{range} + T_{range}, \quad (9)$$

where $Range_{max}$ is the maximum adjustable range for T_{range}.

Each pixel of the aerial image is associated with a 2-bit attribute called color preference. It is set to 1, 2 or 3 if the corresponding channel is dominant or 0 if no channel can obtain the dominate position. A region's color preference is set to be the color preference of the seed pixel. We use a more strict T_{range} value if the current expanding pixel has a color preference different from the growing region.

Those pixels that pass the above three expansion requirements will form the initial ROI. Regions with high confidence should be those clearly distinguished from surrounding background and consequently have small dynamic intensity range (*DIR*). Moreover, one region will have high uncertainty if it contains a large number of in-region edge pixels (*#IREP*). Therefore, we define a region *R*'s uncertainty as:

$$UCT(R) = (1 + \frac{DIR}{range\_len}) \cdot (\#IREP) \cdot \tag{10}$$

A larger region has higher chance to encounter in-region edge pixels. Avoiding this, we compute the uncertainty per pixel (*UPP*) as:

$$UPP(R) = \frac{UCT(R)}{R.area} \cdot \tag{11}$$

Initial segments with comparatively large *UPP* or small size / area will be discarded. The rest are called ROI candidates. *UPP* is also used in the region merging step and the final region matching component.

3.3 ROI Candidate Refinement

The actual number of buildings in the scene is typically less than half the number of ROI candidates because many candidates are false positives such as grounds and roads, and some buildings are over-segmented due to factors such as shadows. The candidate refinement consists of two steps handling the two problems respectively.

First, **learning-based region regulation** is to prune those ROI that are too irregular to become building regions or a part of such regions. For each ROI contour, we construct x and y histograms in the roation-relative frame and compute two attributes measuring their peak strength. Linear Discriminant Analysis is applied to the 5D augmented space to decide a linear boundary, which results a quadratic decision boundary in the original space. Around half of the ROI candidates are pruned by this step. Second, **region merging** is used to iteratively merge those regions that are spatially close to each other (especially when their color preferences are compatible) and form additional interest regions. Only ROI candidates with higher confidences (lower *UPP* attributes) will enter the region merging step because regions with high *UPP* already contain too many.

The outputs of our aerial image ROI extraction are interest regions (ROI_{aerial}) and their contour point lists. We also develop an efficient algorithm to extract ROI_{range} from 3D LiDAR data (not covered in this paper).

4 Region Matching under Different Sensors

Given dominant and most-external ROI contours from both aerial images and 3D range data, we choose to use the shape context [15] as our contour descriptor because as a histogram-based approach, it is able to handle issues like pixel location error well. It can also tolerate various shape deformations (common situation in our case due to imperfect segmentation) while capturing the essence of similarity. Last, shape context generates one descriptor for each contour point, which enables us to establish point-to-point correspondences.

Each ROI's contour points are uniformly sampled to form a contour point list of fixed size (N_{CPL}). We ordered the list in a counter-clockwise manner starting from the point with the smallest y coordinate. Each CPL point j on ROI_i is described by its relative angle difference $\theta_{j,k}$ (to other points k of CPL and k≠j) and logarithm normalized distance $r_{j,k}$ using a log-polar histogram:

$$H_{i,j}(b_\theta, b_r) = \#of\{\theta_{j,k} \in bin(b_\theta) \text{ and } r_{j,k} \in bin(b_r) : 0 < k < N_{CPL_i}\} \quad (12)$$

Scale invariance is achieved by distance normalization (we normalize distances using the size of ROI bounding boxes) and by placing shapes of different scales into histograms with a fixed number of r bins. For rotation invariance [7], tangent vectors are computed at each point and treated as x-axis so that the descriptors are based on a relative frame that automatically turns with tangent angles.

Despite many previous efforts in our ROI extraction stage, over-segmentation and segmentation-leaking can still be observed among ROI_a and ROI_r. Therefore it is still important to allow partial matching (fig. 5) in the region matching stage, achieved by forming partial descriptors in our algorithm. Continuous subsets of the original sampled contour points are used. We re-sample the partial contour and form new partial descriptors. Though imperfectly segmented regions will have better chance of matching through this, adding more descriptors will also enlarge the necessary searching space and raise the distinctiveness requirement. To better handle this trade-off, only those partial contours containing larger number of corners, consequently generating richer and more distinctive partial descriptors will be considered. To further restrict the total number of ROI descriptors, we generally compute partial descriptors only for ROI_r, which are relatively clean and more accurate than ROI_a.

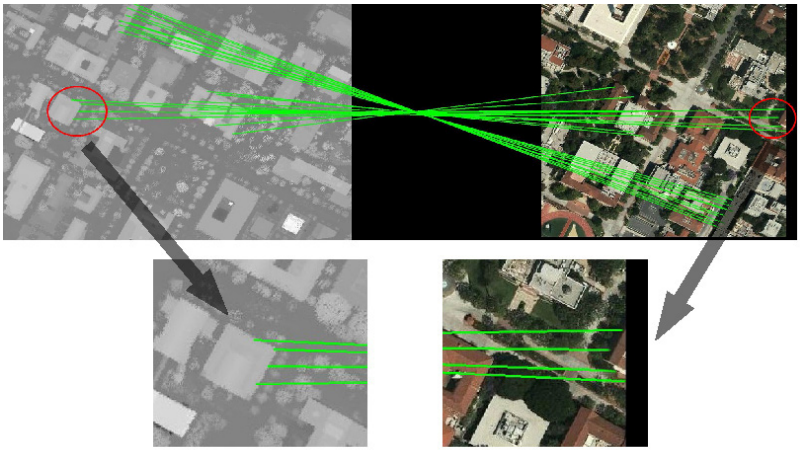

Fig. 3. ROI partial matching

To search for optimal correspondences, for each ROI_r described as N_{CPL} histograms $H_r(j)$, all the $ROI_{a,i}$ ($0 \le i < num_a$) described as $H_{a,i}(j)$ are sequentially scanned. We efficiently measure the similarity of two ROI as the minimum average histogram distance (matching cost) of their corresponding CPL points.

$$\min_{0\le i<num_a}\{\frac{1}{N_{CPL}}\sum_{k=0}^{N_{CPL}}\sum_{j=0}^{N_{CPL}}\frac{H_r(j)-H_{a,i}((j+k)\%N_{CPL})}{H_r(j)+H_{a,i}((j+k)\%N_{CPL})}\} \tag{13}$$

The searching for minimum has a constant low computational cost of $O(N_{CPL})$ because CPL is organized as a counter-clock list of most-external contour points. Once one point's matching is determined, the rest points are automatically corresponded. There is no need to compute the solution for general bipartite matching problem.

After the searching process, each ROI_r is associated with its best and second best matched ROI_a. Among all those tentative correspondences, typically only 10%-40% are correct. The final task is to detect and correct the outliers.

We define "cost ratio" for each ROI_r as the matching cost ratio of its best matching over the second best matching. Lower cost ratio combined with lower *UPP* attributes for ROI indicates a higher matching confidence. For example, regular rectangle buildings are generally ambiguous and produce higher cost ratio because many buildings have similar shapes, while buildings of unique shapes will produce lower cost ratio and higher matching confidence.

For comparatively easy tests with a few distinguished buildings in the scene, correct initial matchings can be found by simply picking several ROI_r with the lowest cost ratio. Each selected ROI_r can contribute 10 uniformly sampled contour points providing a large set of point to point correspondences, based on which a global perspective transformation is estimated using least square method. The result is propagated to those unselected ROI_r using the recovered transformation and produces the final point to point correspondences across the entire scene.

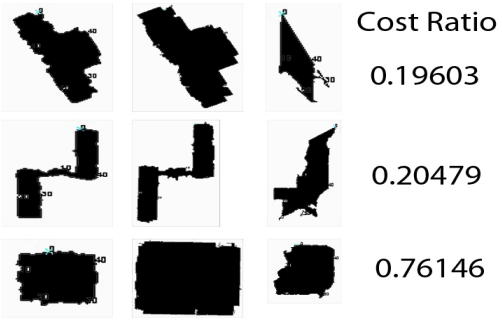

Fig. 4. the 1st column is ROI contours extracted from range data, the 2nd and 3rd column are the best and second best matching from the input aerial image. Cost ratio for each row is given.

For challenging scenes, the correctness of initial matchings can not by solely decided by cost ratio. We propose a unified framework combining outlier removal and matching propagation together. We first construct a subset of matchings with relatively low cost ratio. This high-confidence subset of matchings serves as the foundation group of transformation estimation. For each iteration of the process, we randomly pick one pair of matchings from the subset and compute a global transformation using least square method. The remaining matchings are scanned to locate those consistent with the estimated transformation by comparing the point-to-point correspondences generated by region matching with the matching propagation results. The transformation matrix is updated every time the size of consistent set

increases. We compare and evaluate the results of different iterations using two criteria: the number of propagated matching points that are within the spatial range of ROI_a, and the average *UPP* of the consistent set.

Throughout the process, global context is implicitly taken into consideration. The whole process runs iteratively until a global transformation meeting some predefined criteria is found, in which case matchings have already been propagated to all buildings across the scene, or when the list is exhausted, the system will claim that no correspondence could be established.

5 Experimental Results

The proposed registration method was tested using aerial images and LiDAR data of Atlanta, Baltimore, Denver and Los Angeles. Both real and synthesized data were tested. Though most LiDAR data we used are current and of high resolution, some (e.g. Los Angeles dataset) are captured years ago with very low resolution and some recently built buildings missing from the range data such as the two bottom left buildings in figure 7(c). As a local region based approach, our method can robustly handle such situation common for historic data.

Most aerial images we used are captured in early years with low resolution and from various sources (e.g. returned from online image search engine) and no georeference data can be tracked at all. Others are casually cropped from satellite images. Some testing areas are heavily urbanized with a large number of close buildings while others have sparsely distributed buildings but a lot of vegetations. To focus on different sensors problems in this work, the sides of buildings could be visible but should be comparatively small (Details about how our whole 2D-3D registration system registers images from nadir views to oblique, e.g. [16], are not covered in this paper.). Other than that, we made no assumption about the initial alignment. The images may have any in-plane rotation, even upside down. The scale difference from aerial image to depth image ranges from 0.3 to 3. Perspective and skew distortions could be applied. Concerning location errors, the corresponding building might lie on the opposite corner of the image. The inputs data may originally have no correspondence. Our method is robust enough to handle those factors challenging to general matching and registration system.

Last, the proposed method had been successfully integrated into two application systems for urban rendering and UAV localization respectively.

5.1 ROI Extraction Results

Figure 5 and 6 show the color-coded ROI extraction results from aerial and depth images, compared with results generated by classical segmentation algorithm [10]. Our ROI extraction result meets the particular need of our registration system considerably better than others. The returned segmented regions are more focused on interest buildings and can provide more accurate dominant external contours.

For setting parameters, we choose *UPP* and *OFE* in rather conservative ways only to remove those ROI that are clearly false positives. *T_range* is dynamically related to the current ROI size. We found changing of *range_len* have no significant impact on the segmentation results. Those ROI distinctive from background can robustly be obtained unless some unreasonable values are used, while we were not able to find a

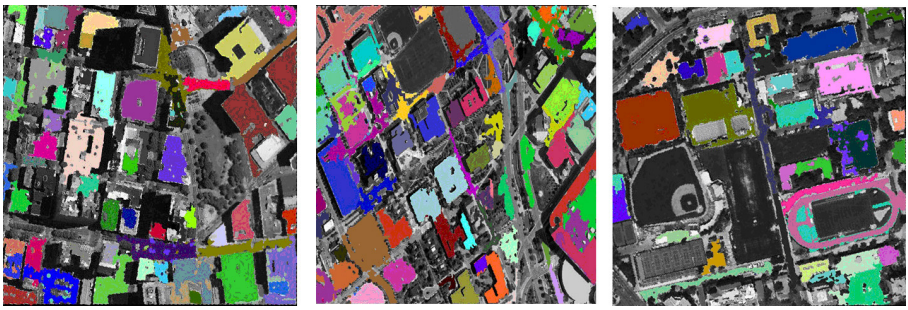

Fig. 5. ROI extraction results (from aerial images)

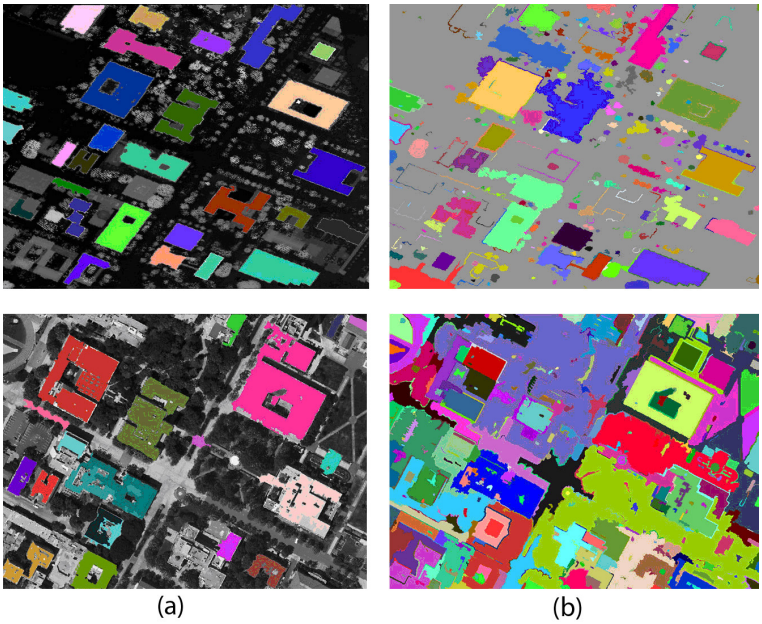

Fig. 6. Segmentation comparison. (a) our ROI extraction algorithm; (b) graph-based segmentation. 1st row for depth image, 2nd row for aerial image.

universal value that can possibly help all the rest ambiguous ones. An average of more than 80% buildings can be correctly extracted from 3D range data during our experiments, while the percentage for correct ROI extraction from aerial images is around 60%. Nonetheless, instead of asking for perfect image segmentation, which is still not feasible today, we also believe the important thing is "how to make the best use of imperfect segmentation results" [8]. In our case, how to establish correct matchings at least for parts of the scene and expand the partial results to the rest.

5.2 2D-3D Registration

First, for registration accuracy, the final average pixel registration error of our method is typically within 5 pixels even for propagated matchings. Methods using high-level

features (better suitable for handling different sensors problem) such as curves and regions typically don't have an accuracy as high as pixel-based methods (e.g. SIFT has sub-pixel level accuracy). That's primarily because of the difficulty of locating exact pixel locations inside high-level features due to many challenging factors, e.g. in our case the influence of shadows, the segmentation leaking and breaking, etc.

Second, like other registration and matching systems, the successful registration of our system also relies on the existence and acquisition of proper matching primitives. In our case, three properly segmented ROI repeated in both 2D image and 3D range data sides are sufficient. This requirement could, sometimes, be difficult to meet either basically because the lack of such primitives in the scene, in which case even human found the registration difficult or impossible, or because such primitives can not be accurately acquired through segmentation technique although it "seems" obvious to human observers.

Our test set currently consists of 918 images, averaging over 200 images for urban areas of each city. Roughly 60% are real images from diverse sources. Large synthesized geometric distortions are applied to those real images to generate the rest. Overall, our method achieves around 56% of success rate for the four city's dataset. To the best of our knowledge, there is no existing registration method that can achieve similar performance without support from positioning hardware. The closest one is: the

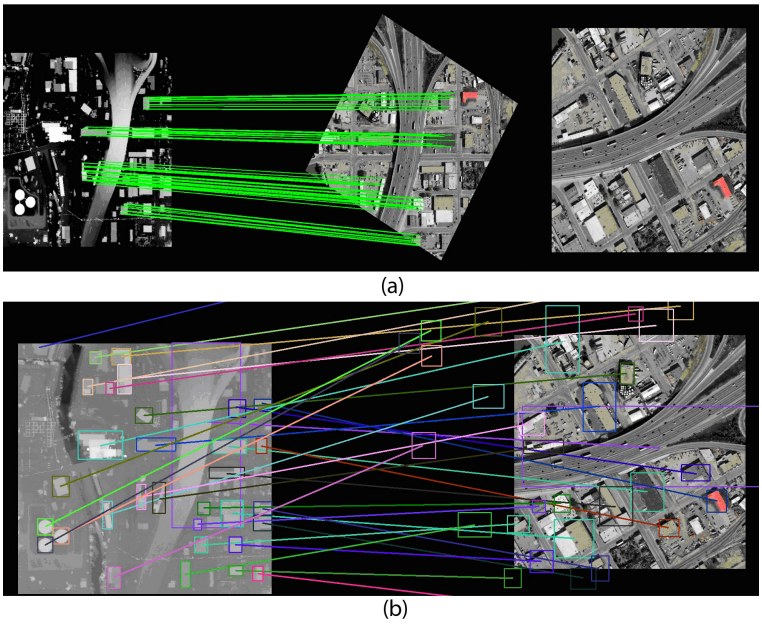

Fig. 7. Registration results of our proposed approach. (a) initial correspondences (left: normalized depth image; right: input aerial image; middle: aerial image wrapped by the recovered transformation); (b) the final results after matching propagation visualized by the bounding boxes and centers of all interest regions' point-to-point correspondences. (c) distorted and partially missing inputs due to historic data. (d) results registering oblique views.

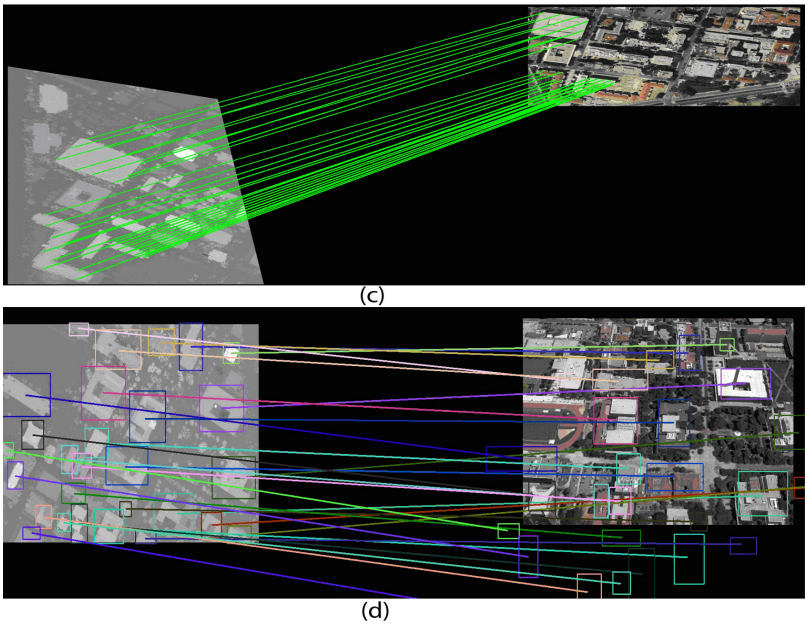

Fig. 7. (*continued*)

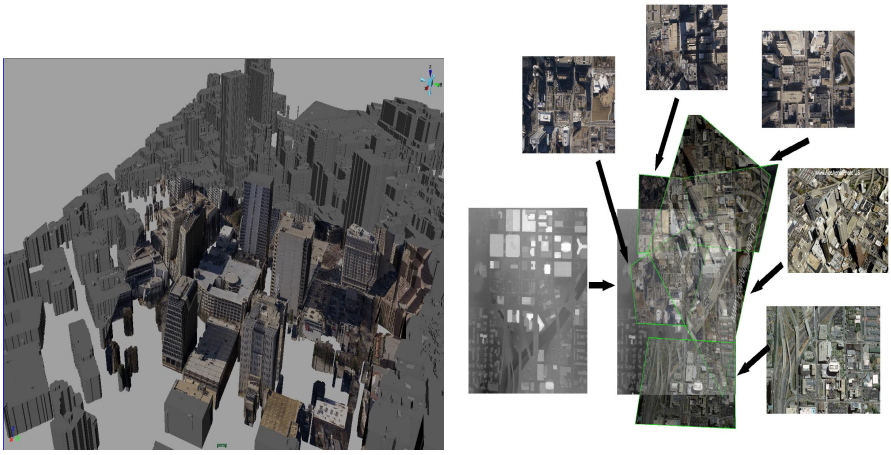

Fig. 8. Apply the proposed approach to urban rendering (left) and UAV localization (right)

system proposed in [5] can directly register 5 camera images out of a test set of 22 images to ground scanned range data. Both methods are working on 2D-3D registration problem without positioning hardware support.

Concerning efficiency, regardless of offline training our entire registration process of one single test for a scene containing around 30 buildings takes roughly one minute in a P4 3.4G PC with a peak memory occupation of 35M.

6 Conclusion

This paper presents our automatic 2D-3D registration method. We provide details for the aerial image ROI extraction component as well as the region matching. Future directions include the propagation of correct registrations to those aerial images that failed the initial registration by iteratively expansion and refinement.

References

1. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (2004)
2. Zhao, W., Nister, D., Hsu, S.: Alignment of continuous video onto 3d point clouds. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8), 1305–1318 (2005)
3. Ding, M., Lyngbaek, K., Zakhor, A.: Automatic registration of aerial imagery with untextured 3D LiDAR models. In: Computer Vision and Pattern Recognition, pp. 1–8 (2008)
4. Fruh, C., Zakhor, A.: An automated method for large-scale, ground-based city model acquisition. International Journal of Computer Vision 60(1), 5–24 (2004)
5. Liu, L., Stamos, I.: Automatic 3D to 2D registration for the photorealistic rendering of urban scenes. In: Computer Vision and Pattern Recognition, pp. 137–143 (June 2005)
6. Liu, L., Yu, G., Wolberg, G., Zokai, S.: Multiview Geometry for Texture Mapping 2D Images Onto 3D Range Data. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2293–2300 (2006)
7. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. Technical Report UCB//CSD-00-1128, UC Berkeley (January 2001)
8. Comanicu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis aind Machine Intelligence 24, 603–619 (2002)
9. Hedau, V., Arora, H., Ahuja, N.: Matching Images under Unstable Segmentation. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Anchorage, AL (2008)
10. Felzenszwalb, P.F., Huttenlocher, D.P.: Effie Graph-Based Image Segmentation. International Journal of Computer Vision 59(2) (2004)
11. Mandelbrot, B.B.: The Fractal Geometry of Nature. W.H. Freeman and Co., New York (1983) ISBN: 0716711869
12. Solka, J.L., Marchette, D.J., Wallet, B.C., Irwin, V.L., Rogers, G.W.: Identification of man-made regions in unmanned aerial vehicle imagery and videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (1998)
13. Wang, Q., You, S.: Explore Multiple Clues for Urban Images Matching. In: International Conference on Image Processing (September 2010)
14. Cao, G., Yang, X., Mao, Z.: A two-stage level set evolution scheme for man-made objects detection in aerial images. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 474–479 (June 2005)
15. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
16. Wang, Q., You, S.: A Vision-based 2D-3D Registration System. In: IEEE Winter Vision Meeting, WACV, Salt Lake City (December 2009)