MATCHING AERIAL IMAGES TO 3D BUILDING MODELS BASED ON CONTEXT-BASED GEOMETRIC HASHING

J. Jung [a, *], K. Bang [a], G. Sohn [a], C. Armenakis [a]

[a] Dept. of Earth and Space Science and Engineering, York University, 4700 Keele Street, Toronto, ON, M3J1P3, Canada - (jwjung, kiinbang, gsohn, armenc)@yorku.ca

Commission I, WG I/3

KEY WORDS: Registration, 3D Building Models, Aerial Imagery, Geometric Hashing, Model to Image Matching

ABSTRACT:

In this paper, a new model-to-image framework to automatically align a single airborne image with existing 3D building models using geometric hashing is proposed. As a prerequisite process for various applications such as data fusion, object tracking, change detection and texture mapping, the proposed registration method is used for determining accurate exterior orientation parameters (EOPs) of a single image. This model-to-image matching process consists of three steps: 1) feature extraction, 2) similarity measure and matching, and 3) adjustment of EOPs of a single image. For feature extraction, we proposed two types of matching cues, edged corner points representing the saliency of building corner points with associated edges and contextual relations among the edged corner points within an individual roof. These matching features are extracted from both 3D building and a single airborne image. A set of matched corners are found with given proximity measure through geometric hashing and optimal matches are then finally determined by maximizing the matching cost encoding contextual similarity between matching candidates. Final matched corners are used for adjusting EOPs of the single airborne image by the least square method based on co-linearity equations. The result shows that acceptable accuracy of single image's EOP can be achievable by the proposed registration approach as an alternative to labour-intensive manual registration process.

1. INTRODUCTION

In recent years, a large number of mega cities provide detailed building models, representing their static environment for supporting critical decision for smart city applications. However, a city is a dynamic entity, which environment is continuously changed and accordingly its virtual models also need to be timely updated for supporting accurate model-based decisions. In this regard, a framework of the continuous city modelling by integrating multiple sources was discussed by Sohn et al. (2013). A first important step for facilitating task is to coherently register remotely sensed data taken at different epochs with existing building models. A large research effort has been made for addressing the problems related to the image registration. A comprehensive literature review can be found in (Brown, 1992; Zitova and Flusser, 2003). Also, Fonseca and Manjunath (1996) conducted a comparative study of different registration techniques using multisensory remotely sensed imagery. Although most of registration methods show promising success in a controlled environment, Zitova and Flusser (2003) also pointed out that the registration is a challenging vision task due to the diverse nature of remote sensing data (resolution, accuracy, signal-to-noise ratio, spectral bands, scene complexity and occlusions). These variable affecting the performance of registration leads to severe difficulty of its generalization. Even though a design of universal method applicable to all registration tasks is almost impossible, the majority of existing registration methods consist of following three typical steps; feature extraction, similarity measure and matching, and transformation (Brown, 1992;

Habib et al., 2005). Thus, a successful registration depends on proper establishment of a strategy for individual steps.

Recently, the advancements of aerial image acquisition technology makes it possible to direct geo-reference. Even though EOPs obtained direct geo-referencing technique provides sufficient accuracy for certain types of applications (coarse localization and visualization), the EOPs obtained though this technique need to be further adjusted for improving their accuracy for many applications where engineering-grade accuracy is concerned including the continuous modelling. Traditionally, accurate EOPs are determined by the bundle adjustment procedure with known ground control points (GCPs) in photogrammetry. However, obtaining or surveying GCPs over large-scale area is labour intensive and time-consuming. An alternative method is to use known points instead of direct survey of GCPs. Nowadays, large-scale 3D building models have been generated over the major cities of the world. Thus, corners of these valuable existing building models can be used for this purpose. However, the quality of building models varies respective to individual building, which is often unknown. Also, computational overhead to match airborne imagery with large-scale building models must be considered for building an effective model-to-image matching pipeline.

To address these issues, we propose a new registration method between a single image and the existing building models. In this study, we propose a new feature which consists of corner and its arms (edged corner feature). In addition to the use of the single feature, context feature is also used to help robust matching. Our matching method is based on the Geometric Hashing method which is a well-known indexing-based object

---

* Corresponding author

recognition technique. However, we rectify the method by introducing several constraints and geometric properties of context feature because a standard geometric hashing method has its own limitations.

1.1 Related Works

Registration process can be recognized to find correspondence between datasets by establishing relation. Brown (1992) classified existing registration methods into area-based and feature-based methods according to their nature. Area-based approach uses image intensity values extracted from image patches. It deals with images without attempting to detect salient objects. Correspondence can be determined with a sliding window of a specific size or over the entire image by correlation-like methods such as; fourier methods, and mutual information methods, and so forth. While, feature-based methods uses salient objects such as points, lines, and polygons to establish relation between two different datasets. The feature-based methods generally consist of feature extraction, feature matching, and transformation. In model-to-image registration, most of registration methods are based on feature-based methods because models have no texture information while salient objects can be extracted from the models and image. Points features such as line intersections, corners and centroids of regions can be easily extracted from models and images. Thus, Wunsch and Hirzinger (1996) used the iterative closest point algorithm (ICP) to register a model to the 3D data. In similar way, Avbelj et al. (2010) used point features to align 3D wire-frame building model with infrared video sequences using a subsequent closeness-based matching algorithm. However, Frueh et al. (2004) pointed out that point features extracted from image cause false correspondence due to a large number of outliers. As building models or man-made objects are mainly described by linear structures, many researchers have used lines or line segments as features for the registration process. Hsu et al. (2000) used line features to estimate a 3D pose of video where the coarse pose was refined by aligning a projected 3D model of line segments to oriented image gradient energy pyramids. Frueh et. al. (2004) proposed model to image registration for texture mapping of 3D models with oblique aerial image using line segment as a feature. Correspondence between line segments was computed by a rating function which consists of slope and proximity. Eugster and Nebiker (2009) also used line features for real-time geo-registration of video streams from unmaned aircraft systems (UAS). They applied relational matching which not only consider the agreement between an image feature and a model feature, but also takes the relations between features into account. However, Tian et al. (2008) pointed out several reasons that make the use of lines or edge segments for registration a difficult problem. First, edges or lines are extracted incompletely and inaccurately so that an ideal edges might be broken into two or more small segments. Secondly, there is no strong disambiguating geometric constraint. While, building models are reconstructed with certain regularities such as orthogonality and parallelism. Utilizing prior knowledge of building structures can reduce matching ambiguities and the search space. Thus, Ding et al. (2008) used 2D orthogonal corner (2DOC) as a feature to recover camera pose for texture mapping of 3D building models. Correspondence between image 2DOC and DSM 2DOC were determined using Hough transform and generalized M-estimator sample consensus. Wang and Neumann (2009) pointed out that 2DOC features are not very distinctive because the feature is described by only orthogonal angle. Instead of using 2DOC, they proposed 3 connected segments (3CS) as a more distinctive

and repeatable feature. For putative feature matches, they applied two level RANSAC which consists of a local and a global RANSAC for robust matching.

2. REGISTRATION METHOD

To register a single image with existing 3D building models, edged corner features are extracted from both datasets and their corresponding matches are computed by an enhanced geometric hashing method. The EOPs of the image can be efficiently adjusted based on the established correspondence of features. The EOPs are updated by an iterative process. Figure 1 illustrates the outline of our approach.
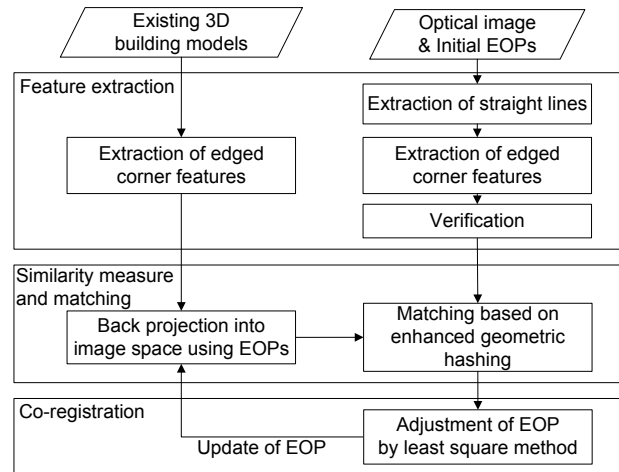

Figure 1. Flowchart of the proposed model-to-image registration method

2.1 Feature Extraction

Feature extraction is the first step of the registration task. The selection of salient features should consider the properties of the datasets used, its application, the required registration accuracy, and so forth. In our study, we use a corner and its arms as a single feature because it can be detected and distinguished in both image and a building model with structure information of a building object. In the building model, it is straightforward to extract edged corner features because each vertex of building polygon can be thought as a corner. In the image with rich texture information, various corner detectors and line detectors can be used to extract the feature. Also, context features are used to achieve more accurate and robust matching results by adding relative geometric information of the context features. In this section, we explain extraction of edged corner features from a single image and properties of context features.

2.1.1 Edged Corner Feature Extraction from Image

Edged corner features from a single image are extracted by three separate steps; 1) extraction of straight lines, 2) extraction of edged corner points and 3) verification of extracted features. The process starts with the extraction of straight lines from a single image by applying a straight line detector. In this study, we used Koversi's algorithm that relies on the calculation of phase congruency to localize and link edges (Korvesi, 2011). Corners are extracted by finding the intersection of extracted straight lines considering proximity with a given distance threshold (T_d = 20 pixels). Corner arms have a certain fixed length (20 pixel) and their directions are determined by two straight lines used. These process may produce incorrect

features because we only considered proximity constraint. Thus, verification process remove incorrectly extracted features based on geometric and radiometric constraints. As a geometric constraint, inner angle between two corner arms is calculated and investigated to remove features with sharp angle. This is because buildings are constructed according to certain geometric regularities (e.g., orthogonality and parallelism) where small acute angles are uncommon. So, features, which have a very acute inner angle (that is, the angle between two arms), are filtered out by a certain inner angle threshold ($T_\theta = 10º$). For applying the radiometric constraint, we analyze radiometric values (digital number (DN) value or colour value) of the left and right flanking regions ($F_1^L, F_1^R, F_2^L, F_2^R$) of corner arms with flanking width (ε) as used in Ok et al. (2012). Figure 2(a) shows a configuration of an edged corner feature and the concept of flanking regions. In a correctly extracted corner, the average DN (or colour) difference between F_1^L and F_2^R, $\left\| F_1^L - F_2^R \right\|$, or between F_1^R and F_2^L, $\left\| F_1^R - F_2^L \right\|$, is likely to be small, underlining the homogeneity of two regions while the average DN difference between F_1^L and F_2^L, $\left\| F_1^L - F_2^L \right\|$, or between F_1^R and F_2^R, $\left\| F_1^R - F_2^R \right\|$, should be large to underline the heterogeneity of two regions. Thus, we measure two radiometric properties, the minimum DN difference of two neighbour flanking regions for homogeneity measurement, $D_{\min}^{homo} = \min\left(\left\| F_1^L - F_2^R \right\|, \left\| F_1^R - F_2^L \right\| \right)$, and the maximum DN difference of two opposite flanking regions for heterogeneity measurement, $D_{\max}^{hetero} = \max\left(\left\| F_1^L - F_2^L \right\|, \left\| F_1^R - F_2^R \right\| \right)$. Thus, a corner is considered as an edged corner feature if the corner has a smaller $D_{\min}^{homo}$ than a threshold T_{homo} and a larger $D_{\max}^{hetero}$ than a threshold T_{heteo}. In order to determine thresholds for two radiometric properties, we assumed that intersection points were generated from both correct corners and incorrect corners, and the two types of intersection points have different distributions of radiometric property. Because there are two cases (correct corner and incorrect corner) for the DN difference values, we can use the Otsu's binarization method (Otsu, 1979) to automatically determine appropriate threshold value. The method is originally designed to extract an object from its background for binary image segmentation based on histogram distribution. It calculates the optimum threshold separating the two classes (foreground and background) so that their intra-class variance is minimal. In our study, a histogram for homogeneity values (or heterogeneity values) of entire intersection points is generated and then the optimal threshold for homogeneity (or heterogeneity) is automatically determined by Otsu's binarization method.
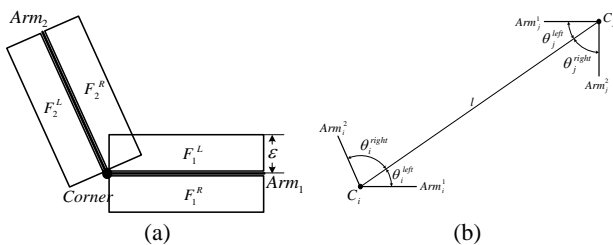
2.1.2 Context Feature

While an edged corner feature provides only local structure information about a building corner, context features partly impart global structure information for configuration of the building object. Context features are set by selecting any two adjacent edged corner features, that is; four angles ($\theta_i^{left}, \theta_i^{right}, \theta_j^{left}$, and θ_j^{right}) between a line (l) connecting the two corners (C_i and C_j) and their arms ($Arm_i^1, Arm_i^2, Arm_j^1$, and Arm_j^2) as shown in Figure 2(b). Note that each angle is determined by the relative line connecting two corners (l). While the context feature is invariant under scale, translation, and rotation, it provides many advantages in the matching process.

2.2 Similarity Measure and Primitives Matching

Similarity measurement and matching process takes place on the image space after existing 3D building models are back-projected into the image space using the co-linearity equations with an initial EOP (or updated EOP). In order to find reliable and accurate correspondences between edged corner features extracted from a single image and building models, we propose an enhanced geometric hashing method where the vote counting scheme in standard geometric hashing is supplemented by a newly developed similarity score function.

2.2.1 Geometric Hashing

Geometric hashing is a model-based object recognition technique for retrieving objects in scenes from a constructed database (Wolfson and Rigoutsos, 1997). In geometric hashing, an object is represented as a set of geometric features such as points and lines, and its geometric relations which are transformation-invariant under a certain transformation. Since only local invariant geometric features are used, geometric hashing can handle partly occluded objects. Geometric hashing consists of two main stages; the pre-processing and recognition stages. The first pre-processing stage encodes the representation of the objects in a database and store them into a hash table. Given a set of object points ($p_k; k = 0,...,n$), a pair of points (p_i and p_j) is selected as a base pair (Figure 3(a)). The base pair is scaled, rotated, and translated into the reference frame. In the reference frame, the magnitude of the base pair equals 1; the midpoint between p_i and p_j is placed at the origin of the reference frame; The vector $\overrightarrow{p_i p_j}$ is the same as a unit vector of x axis. The remaining points of the model are located in the coordinate frame based on the corresponding base pair (Figure 3(b)). The locations (to be used as index) are recorded with the form (Model, used base pair ID, corner index (x and y coordinates in reference frame)) in the hash table which is quantized by a proper bin size. For all possible base pairs, all entries of corner points are similarly recorded in the hash table (Figure 3(c)).

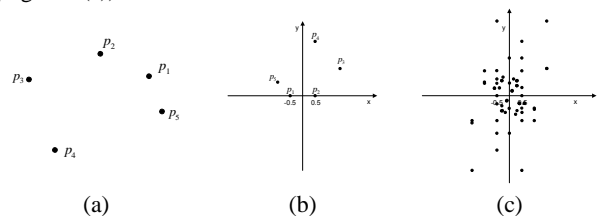Figure 2. (a) Edged corner feature and flanking regions, (b) context feature

Figure 3. Geometric Hashing (a) the model points, (b) hashing table entries with base pair, (c) all hashing table entries with all base pairs

In the second recognition stage, the invariants derived from geometric features in a scene are used as indexing keys to assess the previously constructed hash table for matching with stored models. In a similar way to the pre-processing stage, two points from a set of points in scene are selected as a base pair. The remaining points are mapped to the hash table and all entries in the corresponding hash table bin receive a vote. Correspondences are determined by a vote counting scheme, producing candidate matches. Although geometric hashing can solve matching problems of rotated, translated and partly occluded objects, it has some limitations. The first limitation is that the method is sensitive to the bin size which is used for quantization of hash table. While a large bin size in the hash table cannot separate between two close points, a small bin size cannot deal with the position error of the point. Secondly, geometric hashing can produce redundant solutions because the method is based on vote counting scheme (Wolfson and Rigoutsos, 1997). Although it can significantly reduce candidate hypotheses, a verification step or additional fine matching step is required to find optimal matches. Thirdly, geometric hashing has a weakness in cases where the scene contains many features of similar shapes at difference scales and rotations. Without any constraints (e.g. position, scale and rotation) based on prior knowledge about the object, geometric hashing may produce incorrect matches due to the matching ambiguity. Fourthly, the complexity of processing increases by the number of base pairs and the number of features in the scene (Lamdan and Wolfson, 1988). To address these limitations, we enhance the standard geometric hashing by changing the vote counting scheme and by adding several constraints such as scale difference of a base and specific selection of bases.

2.2.2 Enhanced Geometric Hashing

In our study, we describe the building model objects and the scene by sets of edged corner features. Edged corner features derived from the existing building models are used to construct the hash table in the pre-processing stage while edged corner features derived from the single image are used in the recognition stage. Each building model in the reference data consists of several planes. Thus, in the pre-processing stage, we select two edged corner features, which belong to the same plane of a building model, as base pair. It can reduce the complexity of the hashing table and ensures that the base pair retains the spatial information of the plane. The selected base pair is scaled, rotated, and translated to define the reference frame. The remaining corner points are also transformed with the base pair. In contrast to the standard geometric hashing, our hashing table contains model IDs, feature IDs of the base pair, the scale of base pair (the rate of real distance of base pair), an index for member edged corner features, and context features generated by combinations with edged corner features. Figure 4(b) shows an example of information to be stored in hashing table. Once all entries with possible base pairs are set, the recognition stage tries to retrieve corresponding features based on the designed score function. In order to reduce search space, two corner points from the image are selected as base pair with two constraints; 1) scale and 2) position constraints. As a constraint on a scale, we assume that scales of base pairs from the model and from the image are similar because initial EOP provides an approximate scale of the image. Thus, a base pair from the image is filtered out if the scale ratio between the base pairs from the image and from the model is smaller than user-defined threshold ($T_s = 0.98$). In addition to the scale constraint, possible positions for endings of base pair can be also restricted with a proper searching space which can be determined by

calculating error propagation with the amount of assumed errors (calculated by iterative process) for initial EOP (updated EOP) of the image and models.
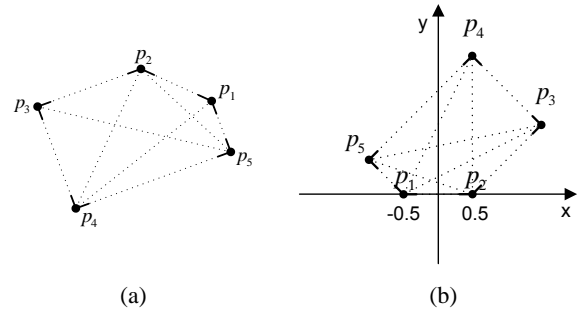

Figure 4. Information to be stored in hashing table (dot lines represent context features to be stored in hashing table.)

After the selection of possible base pair from the image, remaining points in the image are transformed based on a selected base pair. Afterwards, optimal matches are determined by comparing similarity score which combine vote counting scheme and geometric properties of context features. The process starts by generating context features from the model and the image in reference frame. Given a model consisting of five edged corner features (black colour), ten context features can be generated as shown in Figure 5. Note that all corners are not matched with corners from the image (red color). Thus, only matched corners and their corresponding context features (6 long-dash context features in the Figure 5) are used in the calculation of the similarity score function.
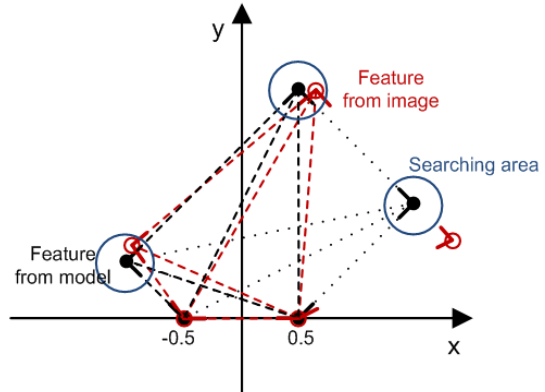

Figure 5. Context features to be used for calculating score function

The newly designed score function consists of a unary term, which measures the position differences of the matched points, and a contextual term, which measures length and angle differences of corresponding context features, as follows;

$$score = \alpha \times \left[w \times \frac{\sum_{i=1}^{n} U(i)}{n} + (1-w) \times \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} C(i,j)}{m} \right] \quad (1)$$

where,

$$\alpha = \begin{cases} 0 & if \ \frac{\# \ of \ matched \ \ features}{\# \ of \ \ features \ in \ model} < T_c \\ 1 & else \end{cases} \quad (2)$$

α is an indicator function where the minimum number of features to be matched is determined depending on T_c ($T_c = 0.5$, at least 50% of corners in the model should be matched with corners from the image) so that all features of the model do not need to be detected in the image; n and m are the number of matched edged corner features and context features, respectively; w is a weight value which balances the unary term and the contextual term ($w = 0.5$).

Unary term: The unary term $U(i)$ measures the position distance between edged corner features derived from the model and the image in a reference frame. The position difference $\left\| P_i^M - P_i^I \right\|$ between an edged corner feature in the model and its corresponding feature in the image is normalized by the distance N_i^P calculated by error propagation.

$$U(i) = \frac{N_i^P - \left\| P_i^M - P_i^I \right\|}{N_i^P} \qquad (3)$$

Contextual term: This term is designed to deal with relationship between neighbour features (that is, context feature) in terms of length and four angles. The contextual term is calculated for all context features which are generated from matched edged corner features. For length difference, the difference $\left\| L_{ij}^M - L_{ij}^I \right\|$ between lengths of context features in the model and in the image is normalized by the length N_{ij}^L of the context feature in model. For angle differences, the angle difference $\left\| \theta_{ij}^{M_k} - \theta_{ij}^{I_k} \right\|$ between the inner angles of a context feature is normalized by the N_{ij}^θ ($N_{ij}^\theta = \pi/2$).

$$C(i, j) = \frac{N_{ij}^L - \left\| L_{ij}^M - L_{ij}^I \right\|}{N_{ij}^L} + \frac{\sum_{k=1}^{4} \left(N_{ij}^\theta - \left\| \theta_{ij}^{M_k} - \theta_{ij}^{I_k} \right\| \right)}{4 \times N_{ij}^\theta} \qquad (4)$$

For each model, a base pair and its corresponding corners which maximize the score function are selected as optimal matches. Note that if the maximum score is smaller than a certain threshold ($T_m = 0.6$), the matches are not considered as matched corners. Once all correspondences are determined, the EOPs of the image are adjusted through space resection using pairs of the object coordinates of the existing building models and newly derived image coordinates from the matching process.

3. EXPERIMENTAL RESULTS

The proposed registration method is tested over the Toronto Downtown datasets provided by ISPRS Commission III, WG3/4 (Rottensteiner et al., 2012). Both reference building models manually digitized by human operator and LiDAR-driven building models reconstructed by Sohn et al. (2012) are used as existing building models, respectively, to investigate effects on modelling errors of used existing models. The image used for the test covers the most of the existing building models (Figure 6(a)). A total of 16 check points, which are well distributed over the image, were used to evaluate the accuracy of the EOPs. From the image, a total of 90,951 straight lines were extracted and then 258,486 intersection points were extracted by intersecting any two straight lines with proximity constraint (20 pixels). Out of these, 57,767 intersection points

were selected as edged corner features where approximately 15% and 60% of intersection points are removed by geometric constraint ($T_\theta = 10°$) and radiometric constraint ($T_{homo} = 26$ and $T_{hetero} = 55$ by Otsu's binarization method), respectively (Table 1). After the existing building models were back-projected to the image using error-contained EOPs, edged corner features were extracted from the vertices of the building models in the image space. As shown in Figure 6, some edged corner features extracted from both the existing building models were not observed in the image due to occlusions caused by neighbour building planes. Also, some edged corner features extracted from LiDAR-driven building models do not matched with edged corner features derived from the image due to their modelling errors. Thus, correspondences between features from the image and from the existing building models are likely to be partly established.
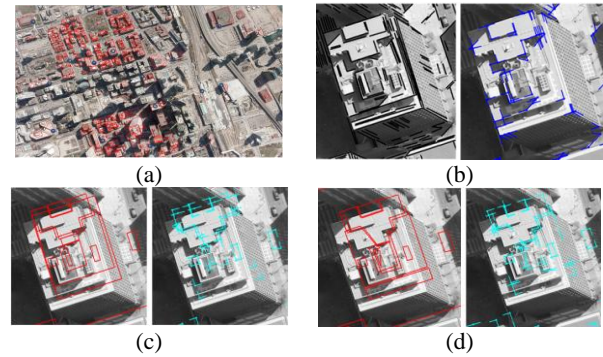

Figure 6. Features Extraction; (a) image and existing building models, (b) image lines (black) and edged corner features (blue) from image, (c) back-projected model (red) and edged corner features (cyan) from manually digitized models and (d) LiDAR-driven models.

The proposed geometric hashing method was applied to find correspondence between features derived from the image and derived from the existing building models. When manually digitized building model is used as the existing building model, a total of 693 edged corner features were matched while a total of 381 edged corner features were matched for LiDAR-driven building models (Table 1). The number of matched features is affected by the quality of used existing building models. Based on matched features, EOPs of the image was calculated by applying the least square method. For qualitative assessment, the existing models were back-projected to the image with refined EOPs of the image. Each columns of Figure 7 and Figure 8 shows back-projected building models with error-contained EOPs (a) and back-projected building models with refined EOPs (c), respectively. In the figures, boundaries of the existing building models are well matched building boundaries in the image with refined EOPs of image.

		# of extracted features	# of matched features
Image	Intersections	258,486	-
	Corners	57,767	-
Existing Building models	Manually digitized building model	8,895	693
	LiDAR-driven building models	7,757	381

Table 1. Extracted features and matched features
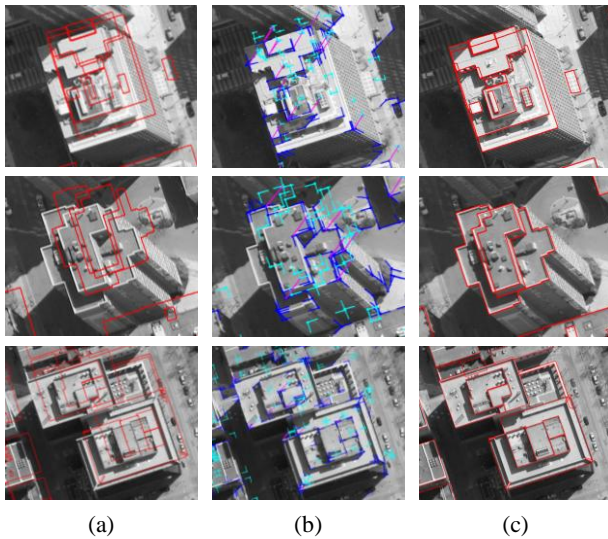
(a)   (b)   (c)

Figure 7. Results with manually digitized building models: (a) with error-contained EOPs, (b) matching relations (magenta) between edged corner features extracted from the image (blue) and the models (cyan) and (c) with refined EOPs
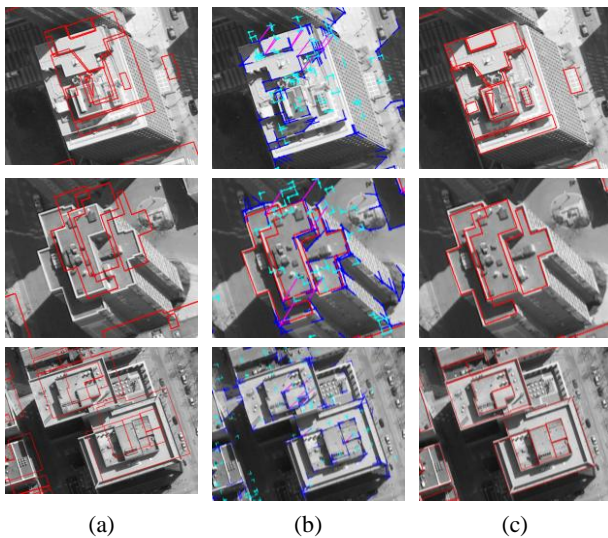

(a)   (b)   (c)

Figure 8. Results with LiDAR-driven building models: (a) with error-contained EOPs, (b) matching relations (magenta) between edged corner features extracted from the image (blue) and the models (cyan) and (c) with refined EOPs

As quantitative evaluation, we evaluated RMSE of check points back-projected to the image space with refined EOPs (Table 2). When $T_m = 0.6$, the result with manually digitized building models shows that the average difference in x and y directions are -0.27 and 0.33 pixels, with RMSE of ± 0.68 and ± 0.71 pixels respectively. The result with LiDAR-driven building models shows that the average differences in x and y directions are -1.03 and 1.93 pixels, with RMSE of ± 0.95 and ± 0.89 pixels, respectively. Although LiDAR-driven building models are used, the accuracy for the check points is less than 2 pixels. Considering that one pixel is approximately 15cm in ground sample distance (GSD), the refined EOPs provides a greater accuracy for engineering applications.

Refined EOPs with manually digitized building models				Refined EOPs with LiDAR-driven building models			
Ave.		RMSE		Ave.		RMSE	
x	y	x	y	x	y	x	y
-0.27	0.33	±0.68	±0.71	-1.03	1.93	±0.95	±0.89

Table 2. Quantitative assessment with check points (unit: pixel)

In this study, threshold, T_m has an effect on accuracy of the EOPs. In order to evaluate the effect on T_m , we measure RMSE of the check points with different values of T_m (Table 3). As T_m is smaller, the number of matched features increase. Interestingly, in terms of accuracy, the T_m was affected by the quality of used building models. When accurately digitized building models are used, the matching accuracy shows constantly good accuracy regardless of T_m . However, the results with LiDAR-driven building models show that the accuracy get worse as T_m is smaller. Also, when high value is assigned to T_m , the number of matched features is too small to recover accurate EOP of image. Therefore, the result indicates that better quality of the existing building models can lead better accuracy of the EOPs.

T_m		0.9	0.8	0.7	0.6	0.5	0.4
Manually digitized building models	# of features	67	268	505	693	766	796
	Ave. x	0.38	0.00	-0.20	-0.27	-0.22	0.25
	y	0.78	0.84	0.31	0.33	0.21	-0.08
	RMSE x	±0.43	±0.81	±0.95	±0.68	±0.81	±1.06
	y	±0.42	±0.97	±1.08	±0.71	±0.66	±0.75
LiDAR-driven building models	# of features	9	98	273	381	438	499
	Ave. x	0.49	-1.09	-1.58	-1.03	-0.43	1.21
	y	-1.93	1.22	1.56	1.93	3.26	2.15
	RMSE x	±7.39	±1.53	±0.68	±0.95	±2.61	±3.06
	y	±6.99	±1.52	±0.61	±0.89	±3.52	±3.66

Table 3. Effects on T_m (unit: pixel)

4. CONCLUSIONS

In this study, we proposed a model-to-image registration method which aligns a single image with the existing 3D building models. Two types of matching cues, edged corner feature and context feature, are proposed for robust registration. From the image, the edged corner features are extracted by calculating intersection of two neighbour straight lines and then verified by geometric and radiometric properties. For similarity measurement and matching, enhanced geometric hashing method was proposed by compensating the limitations of standard geometric hashing method. The qualitative assessment shows that boundaries of the existing building models were aligned with building boundaries of the image using refined EOPs. The quantitative assessment shows that use of existing building models can be applied to find accurate EOPs of image with acceptable and reliable accuracy. Also, an analysis on the effect of the used threshold value used was conducted. The results shows that the more accurate building models is used, the reliable accuracy can be achieved. As future works, we will conduct various analysis to confirm our proposed method's performance in various aspects.

REFERENCES

Avbelj, J., Iwaszczuk, D., Stilla, U., 2010. Matching of 3D wire-frame building models with image features from infrared video sequences taken by helicopters or UAVs, IAPRS, Vol. XXXVIII, Part 3B, pp 149-154.

Brown, L.G., 1992. A survey of image registration techniques, ACM Computing Surveys 24, pp. 326-376.

Ding, M., Lyngbaek, K., Zakhor, A., 2008. Automatic registrationof aerial imagery with untextured 3D LiDAR models, CVPR'08.

Eugster, H., Neibiker, S., 2009. Real-time georegistration of video streams from mini or micro uas using digital 3D city models, 6th International Symposium on Mobile Mapping Technology, Presidente Prudente, Sao Paulo, Brazil.

Fonseca L, M. G. and Manjunath B. S., 1996. Registration techniques for multisensor remotely sensed imagery. PE&RS, Vol. 62, No. 9, pp. 1049-1056.

Frueh, C., Russell, S., Zakhor, A., 2004. Automated texture mapping of 3D city models with oblique aerial imagery. 3DPVT'04.

Habib, A., Ghanma, M., Morgan, M., Al-Ruzouq, R., 2005. Photogrammetric and LiDAR data registration using linear features. PE&RS 71(6), pp. 699-707.

Hsu, S., Samarasekera, S., Kumar, R., Sawhney, H., S., 2000. Pose estimation, model refinement, and enhanced visualization using video, CVPR'00, pp. 488-495.

Kovesi, P.D., 2011. MATLAB and octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia.

Lamdan, Y. and Wolfson, H., 1988. Geometric hashing: A general and efficient model-based recognition scheme. ICCV'88, pp. 238-249.

Ok, A. O., Wegner, J. D., Heipke, C., Rottensteiner, F., Soergel, U. and Toprak, V., 2012. Matching of straight line segments from aerial stereo images of urban areas. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 74, pp. 133–152.

Otsu, N, 1979. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics. 9(1), pp. 62-66.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitex, S. and Breitkopf, U., 2012. The ISPRS benchmark on urban object classification and 3D building reconstruction. ISPRS Annals, 1(3), pp. 293-298.

Sohn, G., Jung, J., Jwa, Y., Armenakis, C., 2013. Sequential modeling of building rooftops by integrating airborne LiDAR data and optical imagery: Preliminary results. In Proceedings of VCM 2013-The ISPRS Workshop on 3D Virtual City Modeling, Regina, pp. 27-33.

Sohn, G., Jwa, Y., Jung, J., Kim, H. B., 2012. An implicit regularization for 3D building rooftop modeling using airborne data, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-3, 2012 XXII

ISPRS Congress, 25 August – 01 September 2012, Melbourne, Australia.

Tian, Y., Gerke, M., Vosselman, G., Zhu, Q., 2008. Automatic edge matching across an image sequence based on reliable points, International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science, 37(Part 3B), pp. 657-662.

Wang, L., Neumann, U., 2009. A Robust Approach for automatic registration of aerial images with untextured Aerial LiDAR data. in Proc. 2009 IEEE Computer Society Conf., CVPR 2009, pp. 2623-2630.

Wolfson, H. J., Rigoutsos, I., 1997. Geometric hashing: an overview, IEEE Trans. Computational Science Eng., vol. 4, no. 4, pp. 10-21.

Wunsch, P. and Hirzinger, G.,1996. Registration of CAD-models to images by iterative inverse perspective matching. ICPR'96.

Zitova, B. and Flusser, J., 2003. Image registration method: a survey. Image and Vision Computing, Vol. 21, pp. 977-1000.