Appl Intell (2010) 32: 216–230
DOI 10.1007/s10489-010-0221-8

Supervised training database for building recognition by using cross ratio invariance and SVD-based method

Hoang-Hon Trinh · Dae-Nyeon Kim · Kang-Hyun Jo

Published online: 10 April 2010

Abstract This paper describes an approach to training a database of building images under the supervision of a user. Then it will be applied to recognize buildings in an urban scene. Given a set of training images, we first detect the building facets and calculate their properties such as area, wall color histogram and a list of local features. All facets of each building surface are used to construct a common model whose initial parameters are selected randomly from one of these facets. The common model is then updated step-by-step by spatial relationship of remaining facets and SVD-based (singular value decomposition) approximative vector. To verify the correspondence of image pairs, we proposed a new technique called cross ratio-based method which is more suitable for building surfaces than several previous approaches. Finally, the trained database is used to recognize a set of test images. The proposed method decreases the size of the database approximately 0.148 times, while automatically rejecting randomly repeated features from the scene and natural noise of local features. Furthermore, we show that the problem of multiple buildings was solved by separately analyzing each surface of a building.

Keywords Cross ratio-based verification · SVD-based method · Supervised training · Building recognition

H.-H. Trinh · D.-N. Kim · K.-H. Jo (✉)
Graduate School of Electrical Engineering, University of Ulsan, Korea, Daehak-ro 102, Mugeo-Dong, Nam-Ku, Ulsan 680-749, Korea
e-mail: acejo@ulsan.ac.kr

H.-H. Trinh
e-mail: hhtrinh@islab.ulsan.ac.kr

D.-N. Kim
e-mail: dnkim@islab.ulsan.ac.kr

1 Introduction

Object recognition is often performed by matching the features of a new image and those of known images. The problems of object recognition such as the size of a database, object extraction, constructed feature, method and constraint of match are challenging tasks because they directly affect the robustness and the rate of recognition.

Those problems are differently considered according to the methods of recognition and the selection of object characters. In the appearance-based method [2, 21, 31], for example, the object extraction is very important because the object would be isolated from the environment. To improve recognition rate, the method is performed with the assumption that the object is handling [2] or appears in the black background. Furthermore, the size of database is usually so large because each object appeared many times with different points of view and scale [2].

The local feature-based method [10–12, 16, 17, 20, 31] is rapidly improved and widely used to recognize the object with the generality, robustness and easy learning [12]. Having many descriptors in the database is the major limitation of the method, this can be explained that one object appears several times and hundreds of descriptors are stored for each pose. It results in several problems such as mismatches and random noise increasing, high computational time and so on. Many techniques were proposed to overcome these problems such as using the nearest neighbor-based constraint for matching [10], choosing the informative local features (iSIFT) [4, 5], selecting the strong identification features [6], probability based selecting of frequently appearing features [31], automatically detecting the objects to reduce the random noise of the scene [27, 28], clustering the similar features as an orderless distribution of features [13, 29, 30]. Here, the facets of a building are first detected by our previous works [24–26, 28] to reduce the random noise of the scene. Then spatial relationship of local features is used to construct only one common model for each building surface. Furthermore, the natural noise of local features is automatically reduced step-by-step when the model is updated by using SVD-based method.

The method and constraint of match also play an important role to reduce the mismatches. For searching the database, a distance is typically evaluated between the test feature and the stored ones. There are three conditional constraints for selecting the best match. The first one is a threshold-based constraint (TBC) in which two features are matched together when their distance is below a certain threshold. So one test feature can get several matches. The second condition is nearest neighbor-based constraint (NNBC) where two features are matched if their distance is not only smallest but also below a certain threshold. With this constraint, a test feature can have only one match. The other one is nearest neighbor distance ratio (NNDR) which is similar to the second constraint. The difference is that a match will be rejected if the ratio between distances of the second nearest neighbor and the nearest neighbor is larger than a threshold [11]. This constraint selects a feature which strongly identifies the object. Remarkably, the third constraint is useful for searching the object in the general environment. It means that only one pose for each object is appeared in the database. But the robustness of recognition will be decreased when the poses of each object that are stored in the database appears several times [19, 31]. Since the appeared poses compete together, and many correct matches are ruled out. In our case, the threshold-based constraint is used to collect as much as possible the repeated features of buildings [6, 18, 31] whose come from the similar building components (BCs) such as the windows, doors or the wall region. The best match of a test image is selected by the largest number of local matches which may include the mismatches [31] or be counted after refining the correct matches [11]. The refinement, the last step of recognition process, is performed by the global transformation with epipolar geometry constraint for general 3D objects, or with homography for planar objects [7]. Recently, the combination of Hough transformation and affine transform is very highly effective to verify the correct matches [11] even there is about 1% inliers. But those methods are not suitable for the objects like building surfaces (Sect. 3.1 is more details).

The proposed method is performed with general scheme as in Fig. 1 where the database is updated step-by-step to automatically reduce the noise within local feature, select the appropriate local feature, reject the noise of the scene; while the size of database is still kept in sufficiently small for each building.

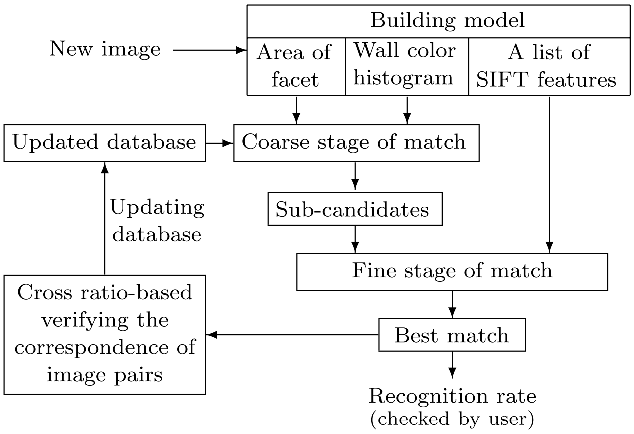


Fig. 1 General scheme for training database

2 Related works

For training database, the accuracy of verified correspondence of image pairs is necessary, therefore we proposed a new technique, called cross ratio-based method whose effectiveness is higher than that of other methods. In order to compare our technique with previous methods in the aspect of application's effectiveness, this section summarizes two popular approaches and the differences between them will be analyzed in Sect. 3.1.

2.1 Canonical RANSAC-based method

This method used RANSAC (Random Sample Consensus, [3]) technique and homogeneous matrices comprising 2D and 3D homography to verify the correspondence of image pairs [7]. In our application, the facets are first detected as planar objects so all processes are performed on 2D coordinate. Therefore, we only consider 2D projective transformation.

Given $N \geq 4$ 2D correspondences $\{X_i \leftrightarrow X'_i\}$, where $X_i = (x_i, y_i, w_i)^T$ and $X'_i = (x'_i, y'_i, w'_i)^T$. A 2D non-singular homogeneous matrix H is determined by $X'_i = HX_i$, or

$$\begin{bmatrix} 0^T & -w'_i X_i^T & y'_i X_i^T \\ w'_i X_i^T & 0^T & -x'_i X_i^T \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = 0; \quad \text{or} \quad A_i \mathbf{h} = 0 \tag{1}$$

where $\mathbf{h}^{jT}$ is the jth row of H, A_i is 2×9 matrix. The error is estimated by

$$e_i^2 = d\big(X_i, H^{-1}X'_i\big)^2 + d\big(X'_i, HX_i\big)^2 \tag{2}$$

$d(X, Y)$ is the Euclidian distance between the inhomogeneous points represented by X and Y.

Firstly, a sample of n ($n = 4$) is randomly selected from N correspondences and H is calculated from this subset by using (1). Secondly the set of data points N_i, whose error is less than a certain threshold, is determined. N_i is the consensus set of the sample and defines the inliers of N. After K trials, the largest consensus set N_i is finally selected and used to re-estimate the matrix H. K is estimated by

$$K = \frac{\log(1-p)}{\log(1-w^n)} \tag{3}$$

where p is a probability that at least one of the random samples is free from outliers after K trials; w is a probability that any selected data point is an inliers. When the information of w is uncertain or this value is small, K is adaptively estimated by pseudo-code of RANSAC algorithm.

2.2 Hough transform-based method

The Hough transform-based method was proposed by Lowe [10, 11]. By using Hough transform and affine transformation, the method can verify the correspondence even the inliers is less than 1%. The method comprises of three steps as follows:

- The NNDR is used; two vectors are matched if their distance is smallest and satisfied, $d_{smallest} \le 0.8 d_{second\_smallest}$. So a test feature has only one match; and all repeated features are discarded.
- The relative parameters comprising 2D location, scale and orientation are used to create a Hough transform entry predicting the closest pose of the test image. Every difference of match such as 30 degrees for orientation, a factor of 2 for scale, and 0.25 times the maximum of training image dimension for location is clustered into one bin of 4D space.
- Each bin with at least 3 entries is verified by affine transformation. The transformation from a model point $(x_i, y_i)^T$ (inhomogeneous coordinates) to a test image point $(x'_i, y'_i)^T$ is described

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{4}$$

where the model translation is $[t_x \; t_y]^T$; the affine rotation, scale and stretch are represented by the m_j ($j = 1, 2, 3, 4$) parameters. This will be written

$$\begin{bmatrix} x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 1 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} x'_i \\ y'_i \\ \vdots \end{bmatrix} \tag{5}$$

Finally, a least squares solution is used to solve the (5).

3 Building property analysis

The building image is represented by the number of facets. Each facet is indexed to the corresponding building and then considered as the independent object. Three characteristics: the area, wall color histogram and a list of local features are used to describe the facet.

3.1 Building is a special object

We now consider the building in urban scene as a special object which results in that the application of proposed method using cross ratio invariance is more suitable than several previous methods.

Building is a big volume of 3D object with similar outer structure and densely appeared in urban scene. Therefore, building is considered as a good landmark when we analyze urban environment [4–6, 18–20, 24, 25]. The outer surfaces contain several components such as windows, doors, wall, columns, balconies, posters and so on. Most of them provide the rich segments in vertical and horizontal directions. We easily detect the building's surfaces by grouping the segments into their dominant vanishing points. Furthermore, building is densely appeared so detecting its surfaces and considering them as independently planar objects are necessary for analyzing the urban scene.

Building is failed in application of canonical RANSAC because of two major reasons. The first one is caused by repeated local feature in the surface where contains rich of similar windows and wall regions, Fig. 2(a). If we use NNDR then many correct matches will be ruled out. On the contrary, when the other constraints are used the number of matches including mis-matches increases so the percentage of inliers become small. This is very sensitive to canonical RANSAC. Figure 2(b) shows an image pair and 104 matches which are obtained by using TBC constraint. In the case of small percentage of inliers, if a mis-sample is large enough, it may become the best result as in Fig. 2(c). In practice, one building may appear several poses in database because of the large changes of illuminance and view points, for example five poses in [19, 31], so the problem of repeated features is more serious. The second reason is resulted from the use of homogeneous matrix that is not proper for the surface of buildings. A rectangular surface, for example, is usually projected into 2D image by a convex quadrangle. But (1) can transform a convex quadrangle to a concave one. Therefore, more mis-samples may get a chance to compete with the correct one, specially in the case of small inliers.

Building is also failed in application of Hough transform-based method by two reasons. The first one is also caused by the use of NNDR constraint. Given an image pair as Figs. 2(e, h), 451 matches are obtained by using TBC whereas only 63 matches are given by NNDR where many

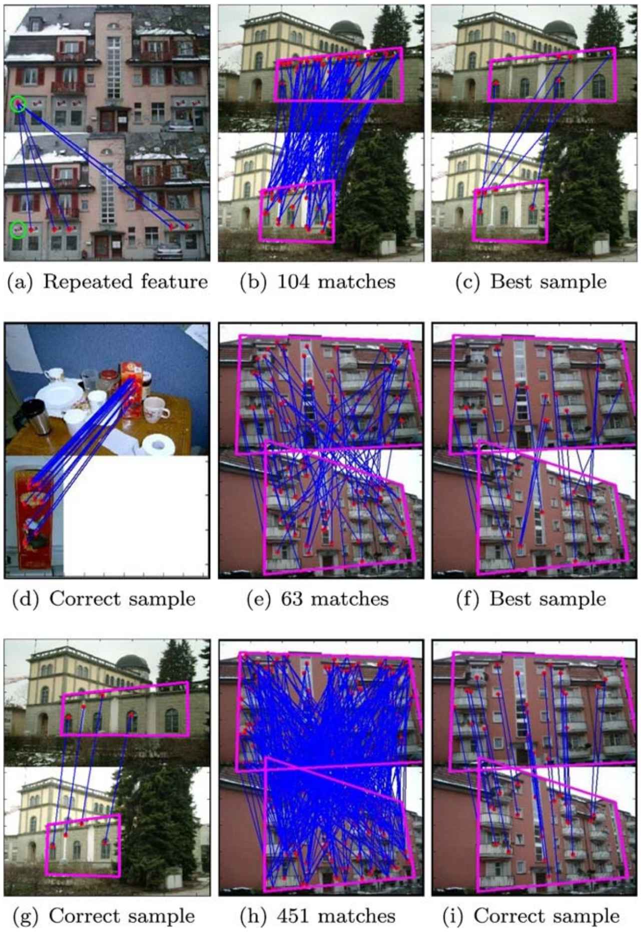


Fig. 2 (Color online) Building is a special object: (**a**) repeated features where *green circles* are the correspondence; (**a**, **b**) failure in canonical RACSAC: (**b**) 104 matches, (**c**) best sample with 6 matches; (**d**) a suitable case for Hough Transform; (**e**, **f**) failure in Hough transform-based method: (**e**) 63 matches with the use of NNDR, (**f**) 21 matches in the largest bin; (**g**) correct sample with 6 matches (several keypoints contains more than 1 descriptor [11]); (**h**) 451 matches with the used of modified threshold-based constraint of proposed method; (**i**) correct sample with 26 matches. The building images are in ZuBuD data [19]

correct matches are rejected. The other reason is explained by the use of affine transformation. Because building is a large object so the affine transformation is not approximate to 2D general transformation. The use of affine transformation, for instance, gives a good result with small object, as in Fig. 2(d), but it should be incorrect with building surface, as in Fig. 2(f) where only 21 matches are found in the largest bin of 4D transformed space (best bin, [11]). Our proposed method would meet the case of building characteristics well (more detail in Sect. 4) so we can get the good results as in the last row of Fig. 2.

3.2 Facet detection

The facet of building images is detected with assumption that the origin of 2D coordinate is embedded at the left bottom corner of image; the axes are coincided with the image boundaries; the apt direction of buildings follows y-axis and the size of images is 640 × 480 (or 480 × 640) pixels. The method of facet detection was explained in detail in our previous works [25, 26]; it contains 5 major steps as follows:

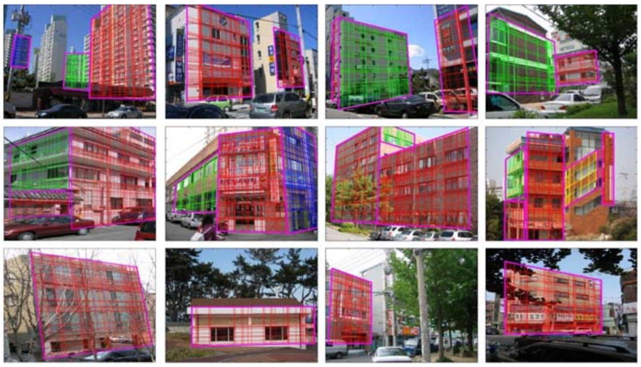


Fig. 3 Examples of facet detection with multiple buildings in image (*first row*), multiple face building (*middle row*), complex environment and general condition (*last row*)

- Detecting line segments from Canny edge detector.
- Segments of non-building pattern are roughly rejected by the contrast of its two neighbor regions.
- MSAC (m-estimator sample consensus) algorithm [3, 22] is used for clustering segments into common dominant vanishing point (DVP).
- The natural properties such as the density and co-existing appearance are used to refine the segments of BCs. And then the vertical segments are extended to across the image.
- The number of intersections between the vertical line and horizontal segments is counted to separate the building pattern into the independent facet. Finally, the boundary of facet as a convex quadrangle is fixed by the empirical conditions.

In Fig. 3, the first row is the result of multiple buildings. Different colors represent the surfaces with different horizontal DVPs. The second row is the multiple face buildings. The last row shows that the proposed method is robust to the complex environment and general conditions. The parameters of DVPs and the facet's area are then used for recognizing a test facet and verifying the correspondence of local features. We will discuss these problems in the next sections.

3.3 Wall color histogram

The facet of building usually contains three major components: windows, doors and wall region. In these components, the color of windows and doors are sensitive. It is caused by several objective reasons such as opened or closed doors,

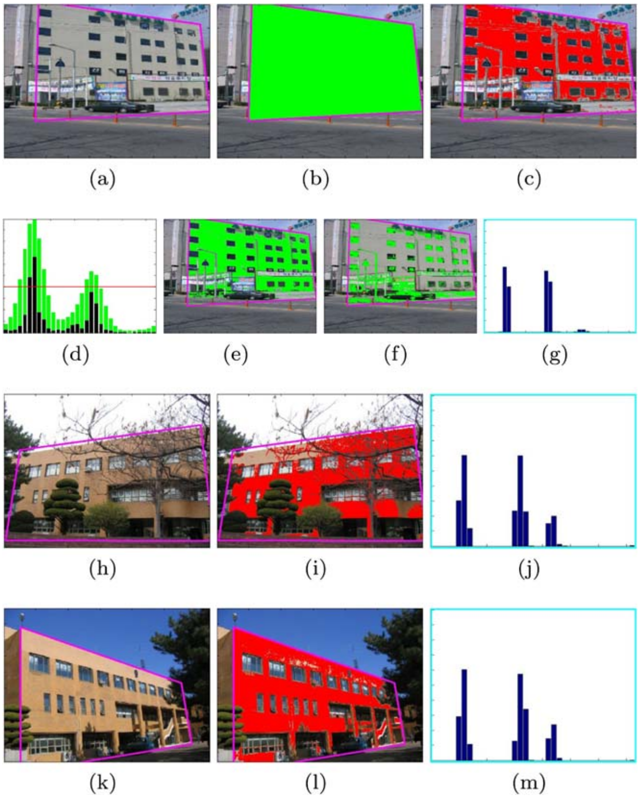


Fig. 4 Wall color histogram: (**a**–**g**) step-by-step illustrate for detecting wall region and its color histogram; (**h**–**m**) the robustness of wall color histogram

the light inside the room, or the glassed windows reflected by the sky light as in Figs. 4(h, k). On the contrary, the wall region is chosen to represent the facet because of less sensitive. Wall region is detected by merging the quadrangles with similar color from the mesh, [28]. Then color information is used to compute a hue histogram with 32 bins. The hue color histogram was successfully used to recognize the building images [27, 28]. A major drawback of hue color histogram is that it is not highly identifiable for gray level of wall regions. Moreover, when the facet contains a huge mesh of quadrangles, the process of searching similar color quadrangles is complex.

We overcome those problems by a new technique. Firstly, all pixels inside the facet's boundaries are extracted, in Figs. 4(a, b). A hue histogram is calculated as the dark graph of Fig. 4(d). Then it is smoothed several times by 1D Gaussian filter. In the experiments, the process is performed with 32 bins for hue histogram, 3 times for smooth and $\sigma = 0.75$ for Gaussian filter. The peaks in the smoothed histogram are detected, and then the continuous bins that are larger than 40(%) of the highest peak are clustered into separate groups. In Fig. 4(d), the light graph is a smoothed histogram, we obtain two separate groups. The pixels indexed by each continuous bin group are clustered together. Figures 4(e, f) show two pixel groups. The hue values are then replaced by gray intensity information and each pixel group is segmented again. Finally, the biggest group of pixels is chosen as wall region as in Fig. 4(c).

Now the information of wall pixels is only used to encode a 36-bin histogram with three components h_1, h_2 and h_3. h_1 (h_2) is a 10-bin histogram of H_1 (H_2) which is calculated by

$$H_1 = \arctan\left(\frac{R}{\alpha G}\right); \qquad H_2 = \arctan\left(\frac{G}{\beta B}\right) \tag{6}$$

where, $0 \leq H_1, H_2 \leq \frac{\pi}{2}$; $\alpha(= 1.1)$ and $\beta(= 1.05)$ are compensation factors due to difference in the change of RGB components under the sunlight. $h_3 = \bar{s}h_{hue}$, where h_{hue} is a 16-bin histogram of hue value; and $\bar{s}$ is the average of the saturation in the HSV (hue, saturation, value) color space. h_1, h_2 and h_{hue} are normalized to unit length. Finally, the wall histogram is created by concatenating h_1, h_2 and h_3 and then again normalized. With gray color of wall region as in Fig. 4(a), the value of $\bar{s}$ is small, so it reduces the affection of hue component (Fig. 4(g)). Whereas, for more specific color, the value of $\bar{s}$ is larger so it increases the effect of hue component like Figs. 4(j, m). Figures 4(h–m) illustrate the robustness of wall histogram. Figures 4(h, k) are two poses of a building under different illumination of sunlight. Figures 4(i, l) are wall regions and Figs. 4(j, m) are corresponding wall histograms. The wall histograms, whose Chi-squared distance (χ^2) is 0.06, are approximate together.

3.4 Rectangular shape and local features of building facets

Among local feature-based methods [8, 9, 11, 14], SIFT (scale invariant feature transform) descriptor gives the good results in most of the test conditions [14]. The keypoints and their SIFT descriptors were proposed by Lowe [10, 11]; they are invariant to image scale, rotation, noise, change in illumination, and specially to the case of occlusion. The SIFT feature is later improved by many authors so that it is adaptable to their applications [1, 4, 5, 8, 14, 31]. We also use SIFT descriptors for our works where the keypoint, which has the larger scale, strongly characterizes the BCs rather than smaller scale [24, 28]. To adapt the natural structures and appearances of buildings, the keypoints whose scale is greater than two is kept for calculating their descriptors [28]. That not only reduces the noises but also saves the time of process. Figure 5(b) draws 2131 keypoints from original image, Fig. 5(a). 191 keypoints, which are satisfied the conditions and located inside the facets, are kept to represent the buildings. Figure 5(c) illustrates selected keypoints and their local regions which are described by the circles. Most of the sizes of local regions are approximate to the size of BCs.

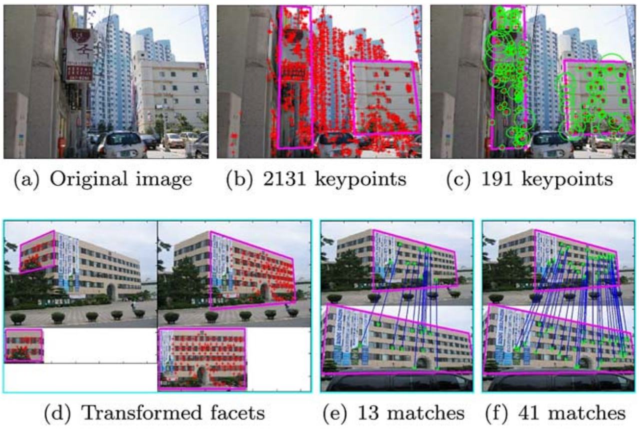


Fig. 5 (Color online) Rectangular shape and local features of building facets; *red* and *green marks* are keypoints; *green circles* approximate to local regions of SIFT features; *blue lines* are the connections of correspondences

In practice, the BCs usually have rectangular shape. When the images are taken from different viewpoints, the images of BCs are distorted into a convex quadrangle region. This problem causes an error that affects to local features [14], so that the recognition rate will be decreased. To overcome this problem, we recover the rectangular shape of detected facet. From the convex quadrangle of facet's boundary, we calculate a rectangle whose length and width equal the average lengths of opposite edges of the quadrangle, respectively. A 2D transformation matrix (H_r) is computed by the quadrangle and the rectangle. Finally, a rectangular shape of facet is recovered by H_r and 2D interpolation method. Now, the distortion of facets is just affected by the scale and stretched transformation, while the rotated affection is remarkably decreased.

The SIFT descriptors are then calculated by the transformed facets. Figure 5(d) demonstrates the building with two detected facets (above). The facets are transformed into the rectangular shapes (below), and then they are used for calculating SIFT descriptors with the below red marks. The above red marks are the keypoints transformed from the rectangular facets to the original faces of building by using 2D transformation matrix (H_r). To demonstrate the effect of using transformed facets, we consider an example as in Figs. 5(e, f). Two images are taken from one building with large scale and rotation. They are directly matched together with threshold constraint. Two descriptors are matched if Chi-squares distance between them is less than 1.5. Figure 5(e) is result when the local features are directly calculated by the original images. We got 13 correct matches after verification. Figure 5(f) shows 41 correct matches when the descriptors are computed on the transformed facets. Then the green marks in Fig. 5(f) are estimated by the corresponding matrix H_r. The results show that the number of correct matches increases approximately three times when we used the transformed facet.

4 Data training

This section describes a technique for training database. Our goals are reducing the size of database, reducing the noise of randomly repeated features coming from the scene, reducing the natural noise by updating the features, rejecting the features which do not repeatedly appear in images taken under different conditions, so that recognition rate is increased. To do so, each building facet is stored in the database by only one model, called common model. Given a set of images of each object, we select the strongly characterized features. The selected features are then transformed to the common model. Because the spatial relationship is used for updating model, a method for exactly verifying the correspondence of image pairs is necessary. Here we proposed a new technique, cross ratio-based method, which is better than several previous methods.

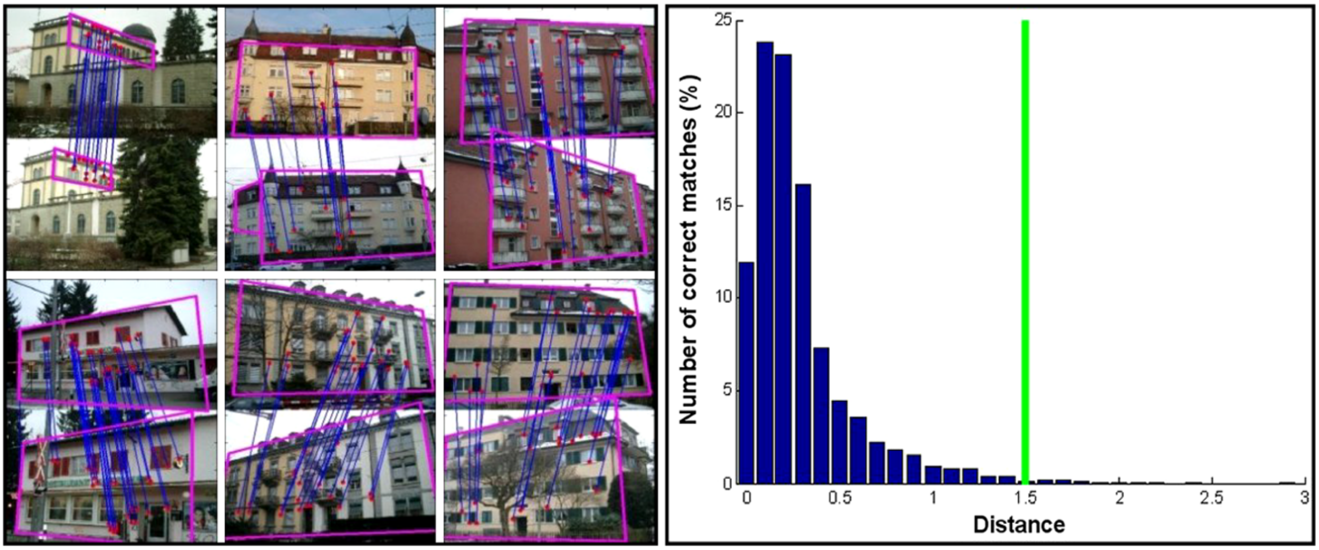
In addition because of repeated features of buildings, if the NNBC constraint is used for selecting the match of a test feature, many correct matches will be refused. So the TBC constraint with slight modification is used to preserve the repeated features of building.

4.1 Matching and constraints

Given a test facet, the recognition progress consists of two stages. Firstly, by using a ratio area and χ^2 distance of wall histogram, a small set of candidates (sub-candidates) is chosen. The thresholds of the ratio area and histogram distances are fixed by $[1/2, 2]$ and 0.1, respectively. Secondly, the recognition is refined by matching the SIFT features. To preserve the repeated structure of building, the TBC constraint with slight modification is used while a test feature can have several matches whose distances are satisfied,

$$\begin{cases} d \le d_0 \\ d \le 1.25 d_{smallest} \end{cases} \tag{7}$$

where, d is Chi-squares distance between the test and the features of sub-candidates. d_0 is a threshold chosen by statistic experiences. For each outdoor object, we took two images in the general conditions of scale, rotation and illumination condition. Each image pair is matched two times with different thresholds τ_1 and τ_2 such that $\tau_1 < \tau_2$. The matches according to τ_1 are used to verify the correspondence. A 2D general projective matrix H is calculated by found correspondences. Finally, H is used to select the correspondences of the matches according to τ_2. The results are considered as the samples for estimating the threshold d_0. In our experiments, τ_1 and τ_2 are chosen by 1 and 5, respectively. Tens of thousands of samples are calculated. Figure 6(a) shows several examples of selected image pairs and computed correspondences. Then the distribution of number of



(a) Examples of sampled correspondences of image pairs (b) Relation of correct matches and their distances.

Fig. 6 d_0 threshold selection by statistics; the images of the examples in (**a**) are taken from ZuBuD data

correct matches according to the distance between matched vectors is investigated as in Fig. 6(b). Here around 99.1(%) number of correct matches is less than 1.5 so the threshold d_0 is fixed by this value.

4.2 Cross ratio-based RANSAC for refinement of local features

To avoid the drawbacks of previous methods, a novel method is proposed by using cross ratio. Our method can refine the correspondence of large planar objects in the case of small percentages of inliers. The cross-ratio of four collinear points A, B, C, D is defined as the "double ratio" [7],

$$\rho_{(ABCD)} = \frac{CA}{CB} : \frac{DA}{DB} \tag{8}$$

If two segments intersect four concurrent lines at A, B, C, D and A', B', C', D' as in Fig. 7(a), then their cross ratios equal together,

$$\rho_{(ABCD)} = \rho_{(A'B'C'D')} \tag{9}$$

We now consider an artificial planar object as shown in Fig. 7(b) with four interest points $\{X_1, X_2, X_3, X_4\}$ and rectangular boundary. Let $\{P_i\}$ points be the projections of $\{X_i\}$ $(i = 1, \dots, 4)$ on the bottom boundary. Therefore, four lines $P_i X_i$ parallel together. Assume that Figs. 7(c, d) are two poses of the above object which are built by general 2D projection. $\{x_i\}$ and $\{x'_i\}$ are the images of $\{X_i\}$ in the left and right poses, respectively. Similarly, $\{p_i\}$ and $\{p'_i\}$ are the images of $\{P_i\}$. In the left pose (Fig. 7(c)), four lines $\{p_i x_i\}$ are concurrent at a vertical vanishing point. Let a, b, c, d be the intersections of $\{p_i x_i\}$, $i = 1, 2, 3, 4$, and the x-axis of the image, respectively. Now, two set of collinear points $\{a, b, c, d\}$ and $\{p_i\}$ are satisfied (9). So we have:

$$\rho_{(abcd)} = \rho_{(p_1, p_2, p_3, p_4)} \tag{10}$$

Similarly we have the following equation for the right image (Fig. 7(d)),

$$\rho_{(a'b'c'd')} = \rho_{(p'_1, p'_2, p'_3, p'_4)} \tag{11}$$

On the other hand, the sets of $\{p_i\}$ and $\{p'_i\}$ are projections of four collinear points $\{P_i\}$. So their cross ratio are invariant [7]. Combining with (10, 11), we have

$$\frac{(x_c - x_a)(x_d - x_b)}{(x_d - x_a)(x_c - x_b)} = \frac{(x_{c'} - x_{a'})(x_{d'} - x_{b'})}{(x_{d'} - x_{a'})(x_{c'} - x_{b'})} \tag{12}$$

Note that, if $x_a > x_b > x_c > x_d$ then $x_{a'} > x_{b'} > x_{c'} > x_{d'}$. This order is considered as a constraint in our method. Given two planar images with available vanishing points and N correspondences $\{X_i \longleftrightarrow X'_i\}$, $i = 1, 2, \dots, N$; Let $\{x_i\}$ and $\{x_{i'}\}$ be projections of the correspondence on the x-axis of each image through the corresponding vanishing points, respectively. From randomly chosen subset of three correspondences $\{A, B, C\}$ and $\{A', B', C'\}$, an error of cross ratio of the ith correspondence following x-axis is defined

$$e_i^x = \frac{(x_i - x_A)(x_C - x_B)}{(x_C - x_A)(x_i - x_B)} - \frac{(x_{i'} - x_{A'})(x_{C'} - x_{B'})}{(x_{C'} - x_{A'})(x_{i'} - x_{B'})} \tag{13}$$

Similar to x-axis, on the y-axis, we get the error e_i^y.

Finally, the correspondences are refined by solving (14) with RANSAC method,

$$\text{minimize} \sum \left((e_i^x)^2 + (e_i^y)^2 \right); \quad i = 1, 2, \dots, N \tag{14}$$

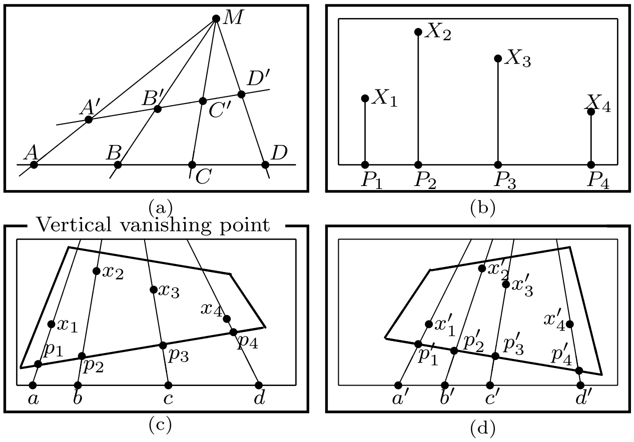


Fig. 7 The illustration of cross ratio-based method: (**a**) concurrent lines; (**b**) artificial object; (**c**, **d**) left and right poses

The experiments show that the proposed method is better than the previous ones in the case of building verification. Figures 2(g, i) are several examples for applying the cross ratio-based method. In Fig. 2(g), 6 inliers (note that several keypoints have more than 1 descriptor [11]) are verified from 104 matches of Fig. 2(b) according to 5.77% inliers. Similarly, there is 5.76% inliers for Figs. 2(h, i).

2D problem is here transformed into 1D problem. The number of loops of RANSAC in our case is smaller than using canonical RANSAC; therefore, the computational time is faster. From (3), the ratio of iteration loops between cross ratio-based RANSAC and canonical RANSAC is calculated by

$$\Gamma = \frac{K_{cross\_ratio}}{K_{canonical\_RANSAC}} = \frac{\log(1-w^4)}{\log(1-w^3)} \tag{15}$$

because the size of sample, n, equals 3 and 4 according to cross ratio-based and canonical RANSAC-based methods, respectively. In our case, the proportion of inliers is usually smaller than 0.15 (15%). By using Taylor formula, the (15) is re-written as follows,

$$\Gamma = \frac{w^4 + \frac{(w^4)^2}{2} + \frac{(w^4)^3}{3} + \cdots}{w^3 + \frac{(w^3)^2}{2} + \frac{(w^3)^3}{3} + \cdots} \approx \frac{w^4}{w^3} = w \tag{16}$$

The number of iteration loops of our method approximates to 0.15 times of canonical RANSAC loops.

In the experiments, local features are calculated after recovering the building facets into the rectangular shape. In this case, the coordinates of keypoints coincide to their projections on the axes. The transformed building facets are related by a simple scale, stretch and translation. Figure 8 shows our method for refining the correspondence of image pairs step by step. The explanation is given in Table 1.

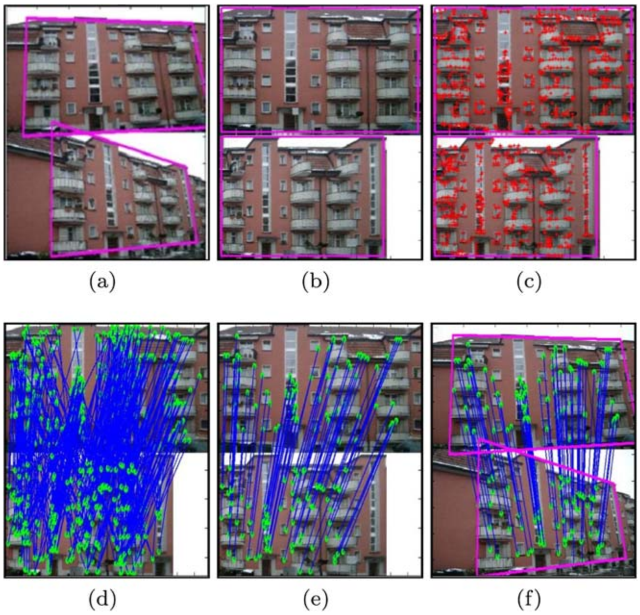


Fig. 8 Step-by-step illustration for searching the corresponding local features of image pairs; (**a**) original images; (**b**) rectangular shapes; (**c**) detected keypoints; (**d**) 518 matches; (**e**, **f**) 90 correspondences

Table 1 Explanation for Fig. 8

Step	Figure	Notes
1	(a)	Two original images with the detected boundaries
2	(b)	Recovered rectangular facets (Sect. 3.4)
3	(c)	Detected keypoits from rectangular facets
4	(d)	518 matches
5	(e)	90 correspondences received by cross ratio-based method
6	(f)	The correspondences are projected into the original images

4.3 Geometric normalization

To improve the results of verification, we implement several techniques for our method. The first technique is for reducing the error caused by far difference values of elements in the same matrix. This problem is affected by the third coordinate of homogeneous coordinates and deeply discussed in [7]. For example, if X_i and X'_i in Sect. 2.1 are given the values $(100, 100, 1)^T$ and $(100, 100, 1)^T$ where w_i and w'_i equal unity. In matrix A of (1), the entries xx',xy',yx',yy' will be of order 10^4; entries xw',yw' etc. of order 10^2; and entries ww' will be unity. When we know the information of image sizes, the error can be reduced by selecting the value of the third coordinates. In report of P.H.S. Torr [23], for instance, w_i and w'_i are chosen to equal 255. The error will now affect to the small coordinates x and y. So we follow the suggestions of R. Hartley et al. [7] in this article. The coordinate of origin is translated to the centroid of the set points. The points are then scaled in order that average distance from the origin equals to $\sqrt{2}$. Assume that this transformation is independently applied for each image , we got $\tilde{X}_i$ and $\tilde{X}'_i$ such that $\tilde{X}_i = TX_i$ and $\tilde{X}'_i = T'X'_i$, respectively. The 2D projective matrix $\tilde{H}$ is received by the set of $\tilde{X}_i$ and $\tilde{X}'_i$. Then the final result is calculated by $H = T'^{-1}\tilde{H}T$. With the normalized parameters, the third coordinates are selected by unity.

The second is for reducing the mis-matches before verifying the correspondences. In the general cases, D.G. Lowe [11] used 4D space with translation, rotation and scale to describe the appeared difference of object from the test and model (training) images. For each dimension, a total of 16 entries was used. This assumption is very good when the size of object appearance is smaller than the size of images. It is a little different for the case of building recognition. The building here is appeared in the upright position with an acute angle 20° in maximum of vertical vanishing point. So that the difference of rotation is selected by 40° for keypoint's orientation. In the experiments, 30° of rotation is a good discrimination capability for training and recognizing images of our data. We also used a factor of 2 for the scale parameter. In our method, the building facets are extracted and considered as independent images. So we have no translation of objects. A relative translation of correspondence is caused by the overlap of detected facets, stretch following x and y axes. The facet's overlap is a common appearance of building face in both of detected facets. The relative translation is described by differences $\Delta\tilde{x}_i = |\tilde{x}_i - \tilde{x}'_i|$ and $\Delta\tilde{y}_i = |\tilde{y}_i - \tilde{y}'_i|$ of correspondence after normalizing. In practice, the area of detected facet is directly proportional to the overlapped region while $\Delta\tilde{x}_i$ and $\Delta\tilde{y}_i$ are inversely proportional to the overlap. For example, if there is no scale and the full face is detected in both of images (maximum overlap), the differences $\Delta\tilde{x}_i$ and $\Delta\tilde{y}_i$ approximate zero. In normal case, the $\Delta\tilde{x}_i$ and $\Delta\tilde{y}_i$ are satisfied the following conditions:

$$\begin{cases} \Delta\tilde{x}_i \le \alpha\Delta_{max\text{-}x} \\ \Delta\tilde{y}_i \le \alpha\Delta_{max\text{-}y} \end{cases} \tag{17}$$

where $\Delta_{max\text{-}x}$ and $\Delta_{max\text{-}y}$ are maximum size of normalized images according to x and y-axis, respectively. α describes the inversely proportional property between the overlap and the area of the test facet as illustration in Fig. 9.

After verification, we get a set of inliers. The 2D projective matrix $\tilde{H}$ is calculated and improved by the set of inliers and using iteratively re-weighted least squares technic [15, 23]. Because the relation between the images now is stretch (scales following x and y-axis are different) and translation (caused by overlap) so the 2D projective matrix, $\tilde{H}$, has a simple form,

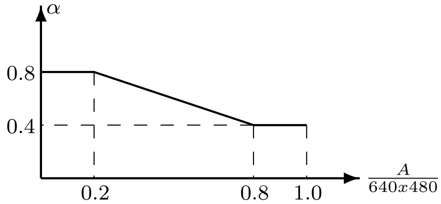


Fig. 9 Relation between factor α and facet's area, A

$$\tilde{X}'_i = \tilde{H}\tilde{X}_i = \begin{bmatrix} \alpha_x & 0 & t_x \\ 0 & \alpha_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \tilde{X}_i \tag{18}$$

where α_x and α_y are scale factors following x and y-axis, respectively.

4.4 SVD-based calculation of the approximative vectors

SVD-based method is widely used to reduce the dimensionality of correlated vectors by selecting their principal components [2, 8]. This means that it filters redundant information. Here, SVD-based method is used for a new application where natural noise caused by objective conditions such as the change of illumination of outdoor scene is considered as the redundant information and will be reduced then.

Given $n \times 2$ matrix A, SVD-based algorithm is used to decompose the matrix A. $A = U\Sigma V^T$, where $\Sigma = \text{diag}(\lambda_1, \lambda_2)$. Let a_1, a_2 be the columns of A, if distance from a_1 and a_2 is too small then $\lambda_1 \gg \lambda_2$ and $\lambda_2 \simeq 0$. In that case, matrix A is replaced by matrix $A' = U\Sigma'V^T$ where $\Sigma' = \text{diag}(\lambda_1, 0)$. Two columns a'_1, a'_2 approximate together. Let $a = a'_1 \approx a'_2$, a is called approximative vector of column a_1 and a_2 because distances from them are very small. If there are more than three similar vectors, we first randomly choose two vectors and calculate the approximative vector. Then this vector is step by step updated by remained vectors. This method is used for updating the extracted features of objects including wall color histogram and SIFT descriptors.

Now, we consider an example given by Fig. 10. Assume that a feature has an ideal sine signal as in the first image of the top row. In reality, the feature is obtained as the second figure because of the noise when images are taken. If the images are taken at different times and points of views, we obtain the extracted features with random noise as the remained figures of the top row of Fig. 10. The noise causes the distance between two features of correspondence greater than zero. And the noise exists in both training and test images. We use SVD-based method here to reduce the noise in the model by a set of training images. The first training image (feature) is chosen as the initial model. Then the model

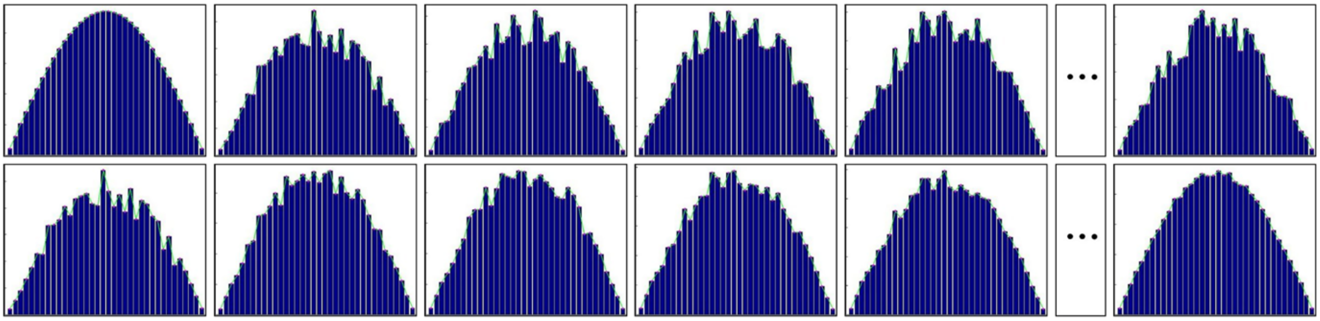
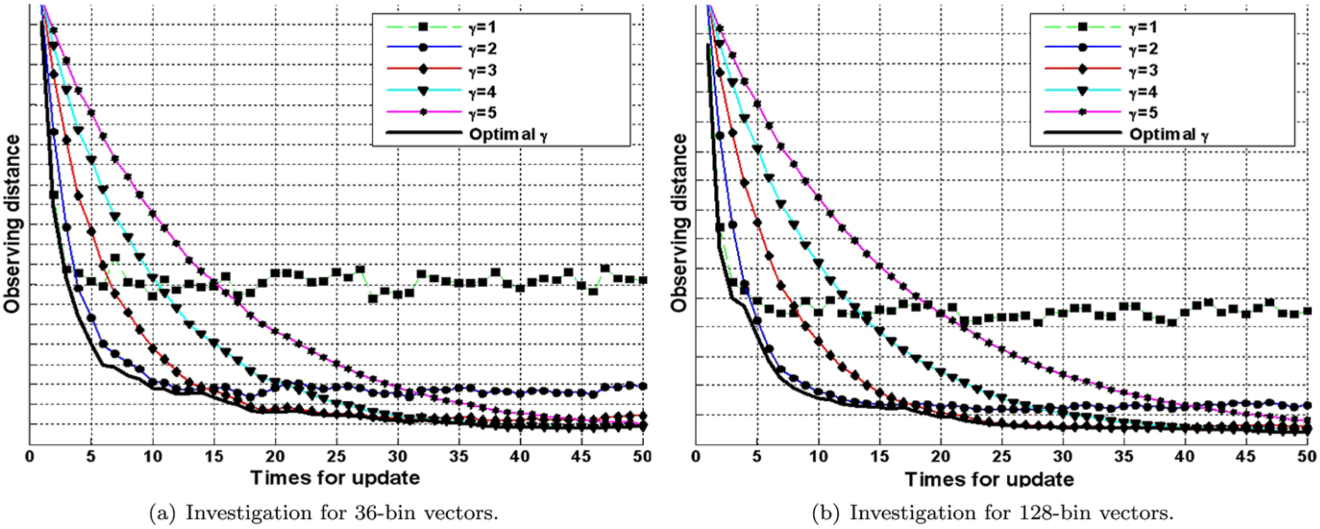


Fig. 10 Automatically reducing the natural noise by SVD-based update of features



(a) Investigation for 36-bin vectors.

(b) Investigation for 128-bin vectors.

Fig. 11 Reduction of the natural noise with different control factors, γ

is updated by the remained training images. The second row of Fig. 10 shows the results of updated models in the step by step. The initial model, updated model, 2, 3, ..., 50 time updated models are shown from left to right. If we call vector a_1 the $(k-1)$th updated model and vector a_2 the kth training feature, the kth updated model is the approximative vector of matrix $A = [a_1\ a_2]$. After 50 times of update, almost noises are rejected, the feature is automatically recovered to the ideal form as in the last figure of the second row. To observe the noise, the distances between the kth updated model and the ideal model is calculated in each step. Furthermore, to improve the robustness of obtained model, matrix A is decomposed with a control factor γ $(A = [\gamma a_1\ a_2])$.

Figure 11 shows the reduction of observing distance following the update times with different γ. The higher value of γ corresponds to the lower affection of the noise, but the recovered time is slower. Figure 11(a) illustrates the experiment with 36 bin vector according to wall color histogram. Figure 11(b) illustrates the experiment with 128 bin vector according to the descriptors of local features. The optimal γ is selected such that the response curve is the under boundary of the curves with various values of γ. According to 36-bin and 128-bin vectors, the γ is selected as in Table 2. In those experiments, the observing distance is reduced and saturated after 20 times for updating. Each value in Fig. 11 is the average of 10 trials with the random signal illustrated as in the top row of Fig. 10. The random signals are estimated by

Table 2 Optimal γ for updating wall color histogram (36 bins) and local features (128 bins)

Size of vectors	Updating times (k)				
128 bins	≤4	[5, 17]	[18, 40]	[41, 60]	>60
36 bins	≤3	[4, 14]	[15, 32]	[33, 50]	>50
γ values	1	2	3	4	5

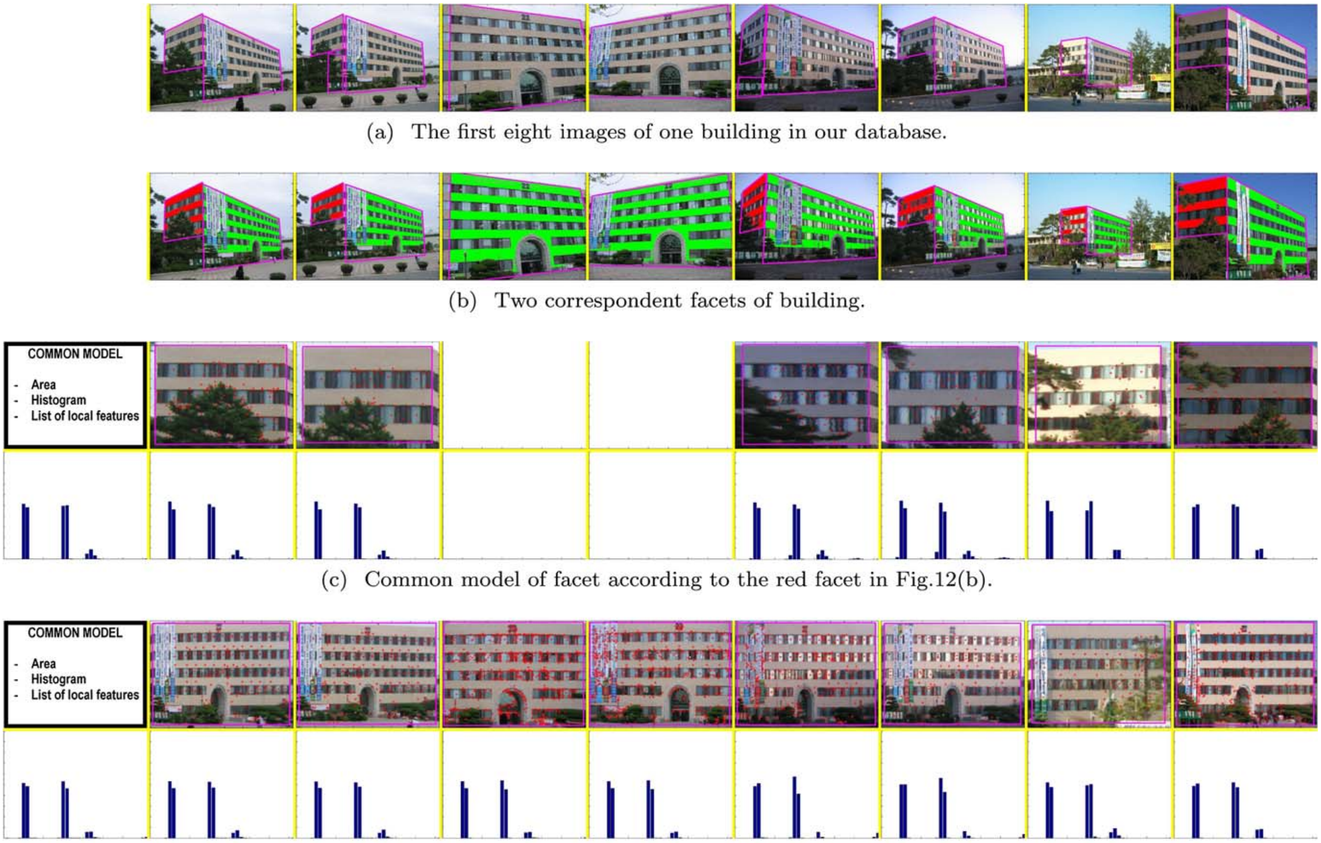
$$x_i^r = x_i^{ISS}(1 + \theta\zeta) \tag{19}$$

where x_i^{ISS} is the ith element of ideal sine signal as in the first figure of Fig. 10 ($i = 1, 2, \ldots, 36$ or $i = 1, 2, \ldots, 128$).

ζ is a random value distributed in the interval $[-1, 1]$; θ is selected such that the average of 1000 trial distances between random signals and ideal signals equals 0.1 and 1.5 according to 36-bin and 128-bin vectors, respectively. The numbers of 0.1 and 1.5 are the thresholds for wall histogram and SIFT descriptors when we recognize the building images (Sect. 4.1). In the experiments, θ is selected by $\frac{1}{3}$ and 0.9, respectively. All the signal vectors are normalized to unit length.

4.5 A common model

This section describes a training method under supervision of user. Each building is represented by number of its surface. For each surface, tens of images are taken under general conditions as the training data set. Then the facets are detected. The corresponding facets are classified by the user. The incorrectly detected facets are ruled out. A common model is constructed for each building's surface by the set of corresponding facets. Each common model is indexed to the building and represented by three properties: the area, wall color histograms and a list of SIFT descriptors. For example, Fig. 12 shows the first eight images of one building in our data set. This building has two faces which can be detected by robot from the road. Total 16 facets are detected by the process. The false positive facets including ambiguities in the second and fifth images are rejected by user. Then, the wall regions of the correct facets are detected. The correspondent facets are also selected by supervision of the user. We have two kinds of correspondent facets whose wall regions are marked by red and green color in the Fig. 12(b), respectively. The wall region is used for calculating the histogram. By using 2D interpolation method, each facet is recovered into the rectangular shape. The recovered facet is used to calculate SIFT descriptions. Figures 12(c, d) show the results of rectangular shape, detected keypoints and wall histogram of the red and green facets in Fig. 12(b), respectively. The red point marks the detector of local features. If a facet does not appear in an image or not pass the area condition, the corresponding data of this facet is represented by the empty space. The example is demonstrated by the third and fourth images of Fig. 12(b) and corresponding data in Fig. 12(c). All histograms of the same surface of a building are used to calculate a histogram of the



(a) The first eight images of one building in our database.

(b) Two correspondent facets of building.

(c) Common model of facet according to the red facet in Fig.12(b).

(d) Common model of facet according to the green facet in Fig.12(b).

Fig. 12 Construction of common model: (**a**) Building images, (**b**) facet detection and its wall region, (**c**, **d**) recovered rectangular shape, SIFT keypoints, wall histogram and common models

common model by SVD-based method. The first columns of Figs. 12(c, d) illustrate the common models and histograms.

For training the local features, we calculate local features of all correspondent facets. We then randomly select one of facets and consider it as the common model with initial local features. The remained facets are used to update the common model. For the first update, a new training facet (F_N) is directly matched to the common model. The correspondences are verified by cross ratio-based method and then used for updating the common model by SVD-based method. The F_N is stored at another place as an auxiliary facet (F_A). For the next update, a new image is also used to update the common model. Then F_N is directly matched to F_A. The correspondences are added into the common model as new local features by using 2D projective matrices. The F_A is replaced by F_N. If the number of features in common model increases, some features whose updated times are smallest (not often appear in the different poses) will be ruled out. Because the difference of density of keypoints on the facets affects the recognition rate [31], so the number of keypoints is proportional to its area. With the size 640 × 480 (or 480 × 640) pixels of images and 700 keypoints for the maximum size of facet, the number of keypoints in each facet is calculated by $N = 700\frac{A}{640\times480}$, where A is the facet's area.

5 Experiments

The proposed method has been experimented by supervised training for the urban image database contained 50 interest buildings and their neighborhoods. For each building, 22 poses are taken under general condition. The first pose is chosen as the initial database. 20 other images are used for training database. The remained one is used to test the algorithm. Our images were taken by CCD cameras of family *PowerShot A570 IS, Cannon*, under general condition and auto mode in 2006 and 2007. The process was performed by the computer of Pentium(R) 4 CPU 3.2 GHz and 1.00 GB of RAM and MATLAB with 7.0 version. The computational time for detecting surface of each image is about 3 s, with about 1.5 s for calculating the features. We trained database in off-line mode.

For each of 50 test images, only the biggest area facet is considered. Here, 78 common models are detected for 50 buildings and their neighbors in the database. The average area of detected facets is 52.73(%) the size of image. So each image contains about 369 keypoints and the database contains 18457 keypoints. That size approximates 0.148 times the database's size of the such work [31]. In that approach, 5 poses for each building are stored in the database and each image contains around 500 keypoints. Table 3 shows the comparison of the database size according to each building between our method and the such work.

Table 3 Comparing the size of database for each building

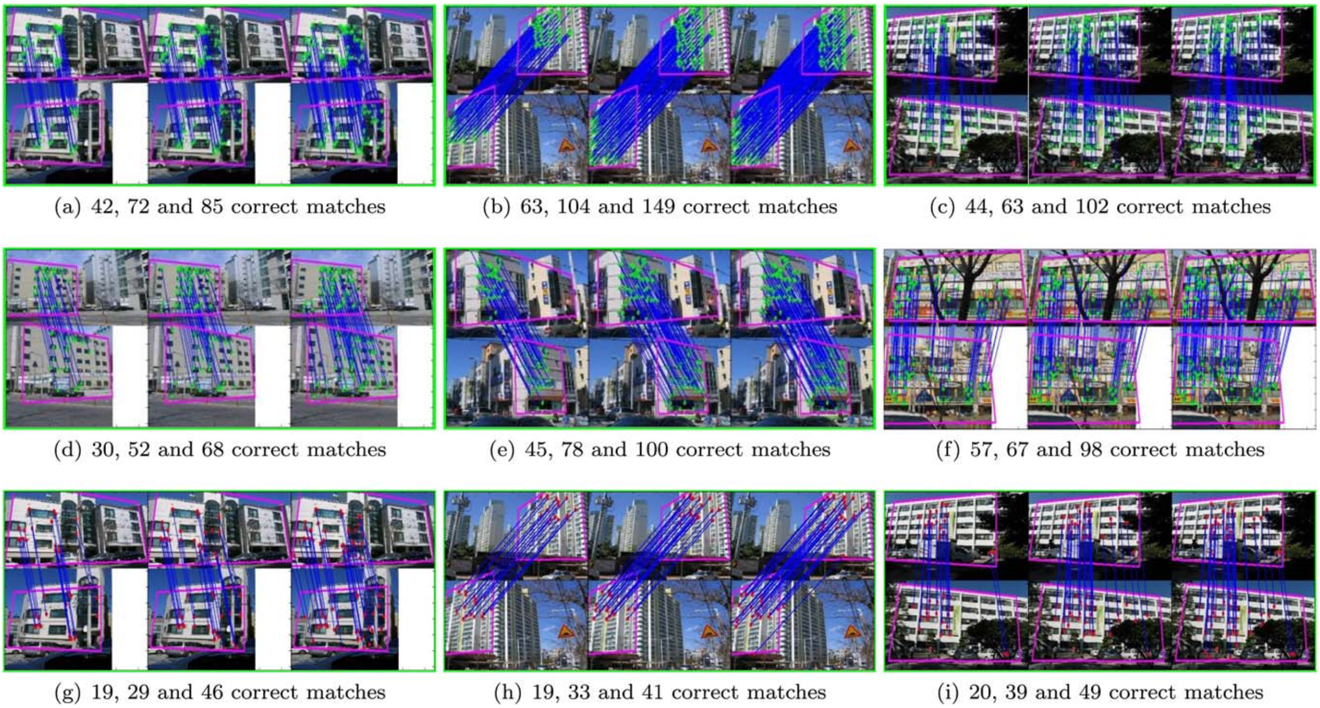
Size of database	Method of W. Zhang [31]	Proposed method
$\frac{\#\text{ images}}{\text{building}}$	5	1 (common model)
$\frac{\#\text{ features}}{\text{image}}$	500	369
Total	2500	369

To observe the effects of updated database and the reduction of natural noise. The test set is performed in every updating step. Two observing parameters are considered. The first one is the largest number of matches between the test facet and updating database. The other is the number of their correspondences. The results show that the largest number of matching increases about 10(%) after each ten times of update. The number of correspondences increases about 50(%) after ten times of update.

Figure 13 shows some examples of our results. For each sub-image, the above image is the test building and the bottom image is the common model whose appearance is represented by the initial image. From left to right, the first, second and third image are the obtained correspondences of without updating, after ten and twenty times updating the database, respectively. We obtain 100(%) recognition rate for the observing test images with the updated database and the average number of sub-candidates is 12.35. Furthermore, the problem of multiple buildings can be solved by separatively analyzing the surface of building. The last row of Fig. 13 shows several examples of the final results without using the recovered rectangular shape of building facets. From the first and last rows of Fig. 13, the obtained matches and correspondences are improved when the recovered rectangular facets are used.

6 Discussion

In this section, we discuss the adaptation of trained model and our future works. We are going to apply the proposed method for designing the outdoor mobile robot for civil and special application. So we are currently investigating to apply the method for unsupervised training database and robot's self localization in urban environment. Generally, the updating method can be used for both of supervised and unsupervised training the database. The supervised training is introduced in Sect. 4.5. In the case of unsupervised training, the process should be run automatically. Consequently, two key questions can be raised up: How can we know whether the building already existed in database or not? And how


(a) 42, 72 and 85 correct matches

(b) 63, 104 and 149 correct matches

(c) 44, 63 and 102 correct matches

(d) 30, 52 and 68 correct matches

(e) 45, 78 and 100 correct matches

(f) 57, 67 and 98 correct matches

(g) 19, 29 and 46 correct matches

(h) 19, 33 and 41 correct matches

(i) 20, 39 and 49 correct matches

Fig. 13 Examples of results, from *left* to *right* in each image: the results of without, 10 and 20 times of update, respectively. The increasing number of matches illustrate that the appropriate features are accumulated in common model. *The first* and *the last rows* show the effect of recovering rectangular shape of building facet

can we be sure that the best match is absolutely correct? Moreover, the training method can be used for the robot to automatically learn the environment and update the database when he is working.

The adaptation consists of automatically selecting the appropriate features, automatically reducing the noise in each local feature as we discussed above. Besides, the trained model can be adaptable to the changes of season or a part of building. Now, we consider the big surface of building in Figs. 12 and 14, Those images of the training set were taken in 2006 and 2007. Certain posters were suspended on one part of building's wall. So they were considered as one part of building and provided several appropriate features for the common model. Assume that the training process was completed, and now the trained models are used with two new images as in Figs. 14(c, d) where the old posters are unraveled. The first new image, Fig. 14(c), is recognized and used to update the model as in Fig. 14(e). Then it is stored as the auxiliary image. Similar performance for the second new image is illustrated in Figs. 14(d, f). We can see that the poster region has not matches because the part of building is changed. In the second step, the new image is directly matched to the auxiliary with 73 correspondences as in Fig. 14(g). Among them, 25 correspondences coincide to the updated features in Figs. 14(e, f); they are shown in Figs. 14(h, i). 48 remained ones are added into the common model as new features. Now, the poster region in trained model, seen as yellow ellipse in Fig. 14(j), is provided the features of the new building's part. If the number of features in common model is increased, the features with small updated time and not recent appearance will be ruled out. When the number of new images is large enough, the old features provided by posters will be replaced by the new ones.

7 Conclusions

A novel method for training the database of outdoor robot is presented. Our method can remarkably reduce the size of database by spatial relation of local features, reduce the random noise from the scene by detecting building facets, decrease the natural noise of local features by using SVD to update the database, increase the correspondence by recovering the rectangular shape of building surfaces, and preserve the effect of repeated structures of objects so the recognition rate is increased. Another contribution of this paper is that the cross ratio-based verification gives the high accuracy of correspondence from image pairs. The method strongly select the correct matches. The transformation of 2D problem to 1D problem reduces the number of RANSAC loops.

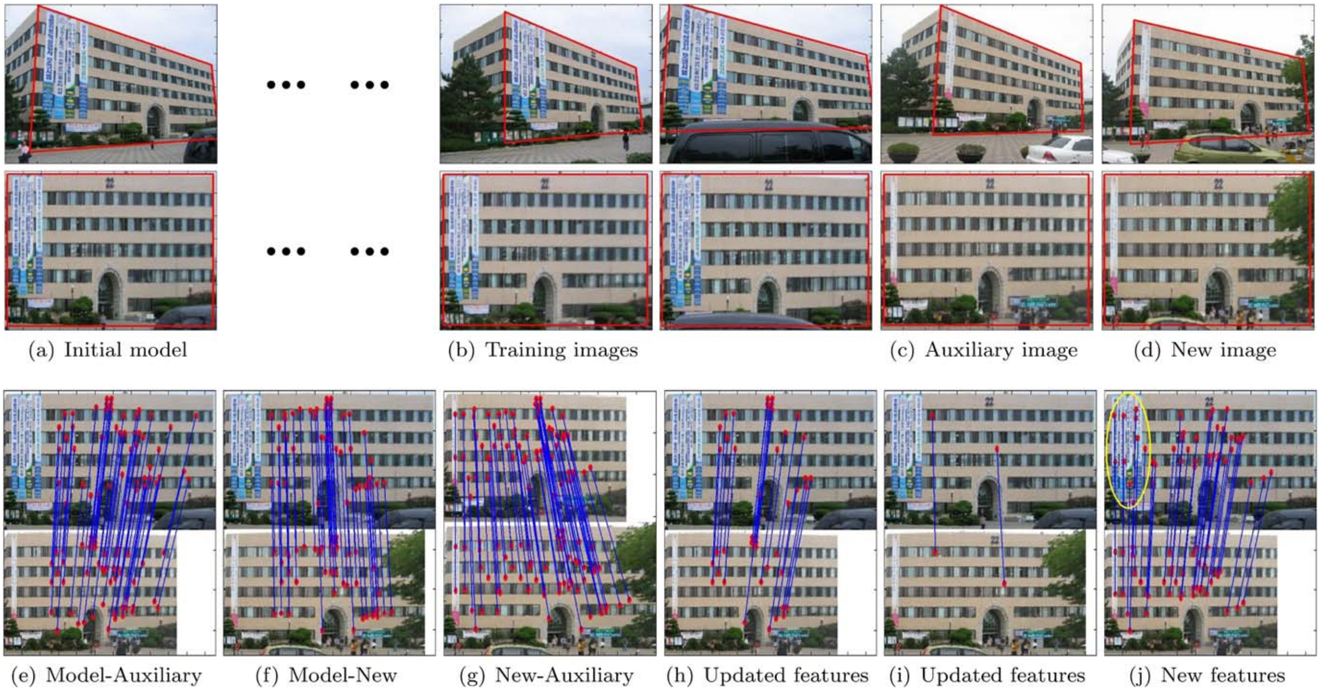


Fig. 14 (Color online) Adaptation of common model: (**a**) image provided initial parameters for common model; (**b**) training images including the images of Fig. 12; (**c**, **d**) new images with assumption that a part of building is changed; (**e**–**j**) illustration of updating process in step by step; *yellow ellipse* is focused region

The proposed method has been applied to recognize 50 building image database which were taken in Ulsan Metropolitan City in Korea. The result of recognition shows to be better than the conventional method while the size of database is approximate 0.148 times smaller.

Acknowledgements This work was supported by post BK21 at University of Ulsan and Ministry of Knowledge Economy/KIAT under Human Resources Development Program for Convergence Robot Specialists.

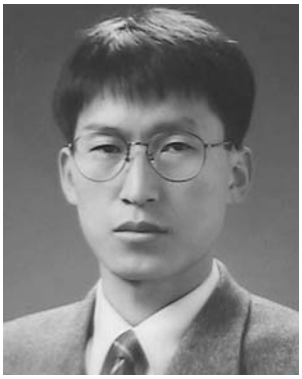
References

1. Bay H, Tuytelaars T, Gool LV (2006) SURF: speeded up robust features. In: LNCS on ECCV, vol 3951, pp 404–417
2. MJ Black, Jepson AD (1998) EigenTracking: robust matching and tracking of articulated objects using a view-based representation. IJCV 26(1):63–84
3. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. Commun ACM 24(6):381–395
4. Fritz G, Seifert C, Paletta L (2005) Urban object recognition from informative local features. In: Proc of IEEE int'l conf on ICRA, pp 131–137
5. Fritz G, Seifert C, Paletta L (2006) A mobile vision system for urban detection with informative local descriptors. In: Proc of 4th IEEE int'l conf on ICVS, p 30
6. Groeneweg NJC, de Groot B, Halma AHR, Quiroga BR, Tromp M, Groen FCA (2006) A fast offline building recognition application on a mobile telephone. In: Proc of IEEE int'l conf on ACIVS, pp 1122–1132
7. Hartley R, Zisserman A (2004) Multiple view geometry in computer vision. Cambridge University Press, Cambridge
8. Ke Y, Sukthankar R (2004) PCA-SIFT: a more distinctive representation for local image descriptors. In: Proc of IEEE computer society conf on CVPR, vol 2, pp 506–513
9. Lazebnik S, Schmid S, Ponce J (2005) A sparse texture representation using local affine regions. IEEE Trans Pattern Anal Mach Intell 27(8):1265–1278
10. Lowe DG (1999) Object recognition from local scale-invariant features. In: Proc of int'l conf on ICCV, pp 1150–1157
11. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. IJCV 60:91–110
12. Matas J, Obdrzalek S (2004) Object recognition methods based on transformation covariant features. In: Proc of European signal processing conf on EUSIPCO
13. Marszalek M, Schmid C (2006) Spatial weighting for bag-of-features. In: Proc of IEEE computer society conf on computer vision and pattern recognition, vol 2, pp 2118–2125
14. Mikolajczyk K, Schmid S (2003) A performance evaluation of local descriptors. In: Proc of IEEE computer society conf on CVPR, vol 02, p 257
15. Rousseeuw PJ, Leroy AM (2003) Robust regression and outlier detection. Wiley InterScience, New York
16. Rothganger F, Lazebnik S, Schmid C, Ponce J (2006) 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. IJCV 66(3):231–259
17. Schaffalitzky F, Zisserman A (2002) Multi-view matching for unordered image sets, or 'how do i organize my holiday snaps?' In: ECCV02, pp 414–431
18. Schindler G, Krishnamurthy P, Lublinerman R Liu Y, Dellaert F (2008) Detecting and matching repeated patterns for automatic geo-tagging in urban environments. In: Proc of IEEE comp soc conf on computer vision and pattern recognition (CVPR)

19. Shao H, Svoboda T, Gool LV (2003) Zubud-zurich buildings database for image based recognition. Swiss FI of Tech, Tech report No 260
20. Steinhoff U, Omercevic D, Perko R, Schiele B, Leonardis A (2007) How computer vision can help in outdoor positioning. In: LNCS on ambient intelligence, European conference, vol 4794, pp 124–141
21. Swets, Weng J (1996) Using discriminant eigenfeatures for image retrieval. IEEE Trans Pattern Anal Mach Intell 18(8):831–836
22. Torr PHS, Zisserman A (2000) MLESAC: a new robust estimator with application to estimating image geometry. CVIU 78:138–156
23. Torr PHS, (2002) A structure and motion toolkit in Matlab "interactive adventures in S and M". Technical report MSR-TR-2002-56
24. Trinh HH, Jo KH (2006) Image-based structural analysis of building using line segments and their geometrical vanishing points, SICE-ICASE
25. Trinh HH, Kim DN, Jo KH (2007) Structure analysis of multiple building for mobile robot intelligence. In: Proc of int'l conf on SICE, Japan
26. Trinh HH, Kim DN, Jo KH (2007) Urban building detection and analysis by visual and geometrical features. In: Proc of int'l conf on ICCAS07, Seoul, Korea
27. Trinh HH, Kim DN, Jo KH (2008) Building surface refinement using cluster of repeated local features by cross ratio. J Lect Notes Artif Intell 5027:22–31
28. Trinh HH, Kim DN, Jo KH (2008) Facet-based multiple building analysis for robot intelligence. J Appl Math Comput 205(2):537–549
29. Willamowski J, Arregui D, Csurka G, Dance CR, Fan F (2007) Categorizing nine visual classes using local appearance descriptors. In: Proc of the 6th ACM int'l conf on image and video retrieval, pp 242–249
30. Zhang J, Marszalek M, Lazebnik S, Schmid C (2007) Local features and kernels for classification of texture and object categories: a comprehensive study. IJCV 73(2):213–238
31. Zhang W, Kosecka J (2007) Hierarchical building recognition. IVC 25:704–716

Hoang-Hon Trinh was born in DongNai, VietNam, in 1973. He received B.E. and M.E. degree from Electrical-Electronic Engineering of HoChiMinh City University of technology, VietNam, in 1997 and 2002 respectively. He received Ph.D. degree from Electrical Engineering Department of University of Ulsan, Korea, in 2008. His research interests include computer vision, pattern recognition, understanding and reconstructing outdoor scenes, designing the outdoor mobile robot for civil and special applications.

Dae-Nyeon Kim received the B.E. and M.E. degree in Control and Instrumentation Eng. from University of Ulsan, Korea, in 2001 and 2003, respectively. He is currently a Candidate of Ph.D. at the Electrical Eng. University of Ulsan, Korea. His research interests include computer vision, human-computer interaction, pattern recognition, object recognition and understanding of outdoor environment and intelligent mobile robot.



Kang-Hyun Jo has graduated and obtained the B.S. from Busan National University and M.S. and Ph.D. from Osaka University, in 1989, 1993, 1997, respectively. He has worked in ETRI (Electro-Telecommunication Research Center) as a Post-Doc. Fellow during 1997 to 1998. Since March 1998, he has joined and now as a Professor, director of Intelligent Systems Lab., Dept. of EE, University of Ulsan. He is serving as the vice dean of College of Engineering. During 2005 (June)–2006 (July), he had stayed in Kyushu University and KIST (Korea Institute of Science and Technology) as a visiting Professor/Researcher. Now he has contributed to organize many international conferences or gatherings. He is an active member of research societies, IEEE, IEICE, IEEK, ICROS, KRS, KIPS, KIISE, KSAE, KMMS etc.