A Line Segments Matching Method based on Epipolar-line Constraint and Line Segment Features

Jie-xian Zeng
School of Software,Nangchang Hangkong University,
Key Laboratory of Nondestructive Test (Ministry of Education) Nangchang Hangkong University,
Nanchang, Jiangxi, China
E-mail: zengjx58@163.com

Sheng Yu
Key Laboratory of Nondestructive Test (Ministry of Education) Nangchang Hangkong University,
Nanchang, Jiangxi, China
E-mail: yu_sheng4105@163.com

Xiang Fu
School of Software,Nangchang Hangkong UniversityNanchang, Jiangxi, China
E-mail: fxfb163@163.com

Chao-xia Li
National Key Laboratory of Science and Technology on Avionics System Integration, Shanghai, China
E-mail:
lizhaoxia2002@yahoo.com.cn

***Abstract*--Line segments are ordinary in industrial scene, accurate line segments matching is a key step for many applications, such as 3-D reconstruction. A matching method based on epipolar-line constraint and line segment features is proposed. Firstly, the points on line segments between image pairs are matched by epipolar-line constraint. Secondly, geometric descriptor and gray value descriptor are used to describe the line segment features, then the two descriptors are combined into a feature vector, and Euclidean distance between vectors is used to achieve fine match. Experiment results show that the proposed method is accurate and fast.**

***Index Terms*--epipolar-line constraint; line segment featrurs; line segments matching**

I. INTRODUCTION

Line features are common elements in industrial objects [1]. The line features are widely used in such fields as image registration [2], 3-D reconstruction and so on. Compared to point features, line features possess the following advantages: Firstly, line extraction and localization are more accurate than that of point for line contains more feature points [3]. Secondly, line reflects the geometric topology better than point for line can describe the boundary structure of object [1].

Line feature matching has drawn lots of attention in the last few years. Accurate line segments matching is a fundamental task and a key step for a variety of computer vision applications, including 3-D reconstruction, motion estimation, target tracking and so on. A typical example is man-made scenes, which mainly consist of line segments, and line matching often becomes an unavoidable step for their 3D reconstruction. Unfortunately, compared to point and region matching, line segment matching is still a challenging task due to various reasons: inaccuracy of line endpoint locations, no strong disambiguating geometric constraint available, lacking of rich textures in line local neighborhood and so on.

Park and Lee [4] proposed a new eigenvector-based line segment matching algorithm, which is invariant to the in-plane rotation, translation, and scale. But this algorithm can only matching between an object and a reference model, and it is sensitive to fracture of straight line. The improved matching algorithm in [5] solved this problem effectively. Firstly, three line segments are selected as a reference model, and the candidate model is constructed based on the orientational, positional and physical constraints. Then the exact matching model corresponding to the reference model is determined based on the eigenvector technique. Unfortunately, both the two methods require a reference model, and can only match the line segments in the model. Lourakis et al. [6] presented an approach using the “2 lines + 2 points” projective invariant for line matching in images of planar surfaces, hence their method is limited to planar scenes. Herbert [7] proposed a method for automatic line matching in color images, where an initial set of line

doi:10.4304/jsw.6.9.1746-1754

segment correspondences were generated using color histogram, then a topological filter was used to iteratively increase possible matches. The main draw back of this method is its heavy reliance on color rather than purely on texture. While color provides a very strong cue for discrimination, it may fail in the case where color feature is not distinctive, such as in gray images or remote sensing images. Chen [8] and Li [9] proposed matching methods based on straight line segment support region. These methods could not only match line segments accurately for short baseline matching, but also has good effect for wide baseline images. But the line support region is difficult to be determined accurately and the weighting factors for different features need setting manually. Although grouping matching strategy [10] has the advantages that more geometric information is available to removing ambiguities, and is able to cope with more significant camera motion, it often has high computational complexity and is sensitive to line topological connections or inaccuracy of endpoints. Wang [11] introduced the concept of line *r*-parallel region into line segments matching. The gray value of each line segment within the line segment *r*-parallel region is used to calculate a feature vector. And the Euclidean distance between vectors is computed to determine whether line segments are matching or not. However, this method ignored the weight factors for different lines. Actually, lines at difference positions have different contribution to describe the feature line.

In this paper, a binocular stereo vision system is constructed by two CCD cameras in the same horizontal line. An object is shot by the two cameras at the same time, and a right image and a left image are obtained. The line segments for two images will be detected and matched. To avoid the deficiencies of the traditional line matching methods that need reference model and the weighting factors have to be set manually.

Based on the epipolar-line constraint and the descriptors of line segment, a line segments matching method is proposed in this paper. Firstly, epipolar line constraint is used to reduce the search area; secondly, geometric descriptor and gray value descriptor are used to describe the line segment features. A Gaussian weighting function is used to assign weight for each line segment in corresponding *r*-parallel region. Finally, those two descriptors are combined into a feature vector, and Euclidean distance between vectors is used to achieve fine match. Experiment results show the approach is accurate and fast.

The rest of this paper is organized as follows. Section II discusses line segments extraction and description. Line segments are extracted by Extended Hough Transform, which has ability of detecting any line segment with desired length and endpoints precisely. In Section III, two kinds of descriptors of line segment are presented. Section IV presents line segments matching. Section V contains some experimental results and analyses. The conclusion and further extensions are discussed in Section VI.

II. LINE SEGMENT EXTRACTION

Fast and accurate line segment extraction is a crucial step to achieve line segments matching. The common line extraction methods include least-squares approximation, phase grouping method, Hough Transform (HT) and some improved Hough transform methods. In this paper, the Extended Hough Transform (EHT) in [12] is used to extract line segments.

A. *Principles of the Hough Transform (HT)*

The HT is a powerful technique for the determination of parameterizable straight lines in binary images. It is essentially a voting process where each feature point voter for all possible lines passing through that point. The votes are accumulated in an accumulator array, and the maximum vote is considered to be the straight line in the image. The advantage of the transform is its robustness.

Point-line duality is the basic idea of the Hough Transform. In the image space, collinear points of intersection corresponding to a sine curve in the parameter space. The other hand, in the parameter space where all the curves intersect at one point, but in the image space has a corresponding collinear point. So, HT to detect lines in the image space is transformed into detect a point in the parameter space. Through cumulative statistics the number of points in the parameter space could achieve line detection.

As shown in Fig 1(a), in the x-y coordinate system, we consider a point $A(x_i, y_i)$ and a straight line l of the inclined cutting equation $y_i = \alpha x_i + \beta$. Through the point $A(x_i, y_i)$ has many lines, and for different values of α and β, which are to satisfy the equation $y_i = \alpha x_i + \beta$. However, in $\alpha - \beta$ parameter space we rewrite the equation into the form of $\beta = -\alpha x_i + y_i$, and will be have only one straight line equation satisfy point $A(x_i, y_i)$. This situation is illustrated in Fig 1(b). In the parameter space, the second point $B(x_i, y_i)$ also associated with a straight line, and this straight line will be intersect with the first line at a point $C(x_i, y_i)$, which α_l and β_l are the line l's slope and intercept respectively in $x - y$ coordinates.

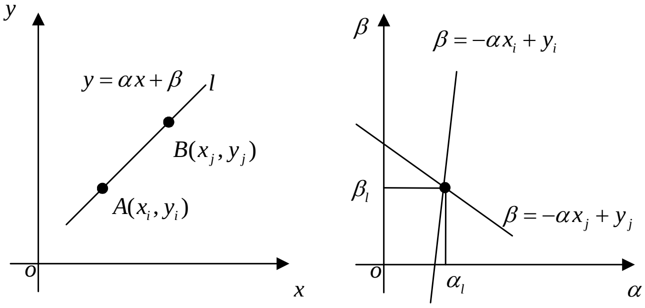
As shown in Fig 2, HT parameter space is divided into accumulator units, where $(\alpha_{\max}, \alpha_{\min})$ and $(\beta_{\max}, \beta_{\min})$ are slop and intercept values respectively of the desired range. In the unit coordinate (i, j) has accumulated value $A(i, j)$. Firstly, all accumulator units are set to zero. Secondly, each point of the image is performed the parameter transform to calculate the point corresponds to some straight lines in the $\alpha - \beta$ parameter space, and add one in the corresponding unit.

That is $A(\alpha_i,\beta_i)=A(\alpha_i,\beta_i)+1$. Finally, the units where accumulator values greater than a threshold value are found, each parameter pair of α and β of a unit correspond to a straight line detected.

However, when a straight line is close to vertical, the slope close to infinity and the intercept in y-axis will also become infinitely. This will lead to lots of problems during accumulator division in the parameter space. So, the polar coordinates are often used to represent straight line, that is:

$$\rho = x\cos\theta + y\sin\theta \tag{1}$$

where ρ is the distance from line l to the origin O, θ is the angle between straight line l and the x-axis.



(a) $x-y$ coordinate system (b) $\alpha-\beta$ parameter space

Figure 1. Principle of HT.

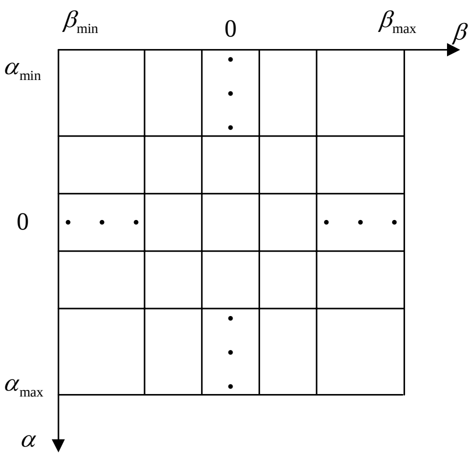


Figure 2. Parameter space division.

B. *Extended Hough Transform(EHT)*

HT is a popular straight line extraction tool for robustness to noise, but it can not get the endpoints of straight line and the computation cost is high. J.Cha [12] proposed EHT, which extended traditional 2-D parameter space to 3-D parameter space. The most distinctive advantage of EHT is the ability of detecting any line segment with desired length and endpoints precisely.

In EHT method, the *x* or *y* coordinate value of the image point with two parameters of the HT constitute the 3-D parameter space. In this paper, the *x* is used as the third parameter, that is, a point in 3-D parameter space of the EHT is represented as (α,β,x). Firstly, the input image with size $h\times w$ is transformed into edge map by using edge detection operator, the canny operator for example. For each pixel, the voting process is performed in two 3-D Hough spaces to detect line segment, each space is consisted of a set of 2-D Hough planes. In the EHT method, each 2-D Hough plane is used to collect the evidence of the line segments which is passing through a specific column of the input image. Considering the first column of the edge map, the edge pixel is selected one by one in a top-down manner. For each selected edge pixel with location $(x,y), 0\le x\le w-1$, and $0\le y\le h-1$, a slope-intercept equation $\beta=-\alpha x+y$ is created, and then the voting process is performed in the first Hough plane of the first Hough space for $-1\le\alpha\le1$ corresponding to the angle range $(-45^\circ,45^\circ]$; after rotating *x* and *y* by 90°, i.e. mapping the edge pixels with location (x,y) into the one with location $(-y,x)$, the voting process is performed in the first Hough plane of the second Hough space. Note that each edge pixel in the first column of the edge map must perform two voting processes in the first and second Hough space respectively. For each column from the second to the last of the edge map, the above two voting processes are applied to the second Hough plane of the first Hough space and the second Hough space.

After finishing the voting processes for the last column of the edge map, we sum up the number of votes for each pair of α and β, if the number of total votes is larger than the specific threshold, it can be claimed that the input image has a line segment with the parameters α and β, its starting point and ending point can be determined by checking the number of votes of each Hough planes.

III. LINE SEGMENT DESCRIPTION

Similar to creating point descriptor, how to select and partition line local neighborhood is the first step to construct our line descriptor. In this paper we improved a scheme to line segment parallel region of different-length lines into uniform description vectors.

A. *Line segment descriptor of mean gray value*

Firstly, the concept of line segment parallel region [11] is introduced as follows. As shown in Fig. 3(a), L is a line segment with length LP in the image plane. Define L as the central axis and $2r$ as the width of a rectangular region, which is called as the *r*-parallel region of line segment L, denoted by $G_r(L)$. Therefore, $G_r(L)$ can be expressed as a set, which consisted by $2r+1$ parallel line segments with equal length, and there is $G_r(L)=\bigcup_{i=1}^{2r+1} l_i$.

In order to make the $2r+1$ parallel line segments each has uniqueness property, the gradient direction of the line segment is defined as the direction of the parallel region, As shown in Fig. 3(b), the direction is marked as $d_{\perp}$. Arrange along $d_{\perp}$, the *r*-parallel region of line segment L has a unique decomposition. That is:

$$Q_r(L) = \{l_1, l_2, \cdots, l_r, l_{r+1}, \cdots, l_{2r+1}\} \tag{2}$$

Thus the *r*-parallel region of line segment L is decomposed to $2r+1$ parallel line segments which have equal length. Point j on line segment i has gray value f_i^j. Gray values of all points in the parallel region are arranged to a matrix form. A $(2r+1)\times LP$ matrix $\mathbf{F}(L)$ can be obtained as follows:

$$\mathbf{F}(L) = \begin{bmatrix} f_1^1 & f_1^2 & \cdots & f_1^{LP} \\ f_2^1 & f_2^2 & \cdots & f_2^{LP} \\ \cdots & \cdots & \cdots & \cdots \\ f_{2r+1}^1 & f_{2r+1}^2 & \cdots & f_{2r+1}^{LP} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^1 & \mathbf{f}^2 & \cdots & \mathbf{f}^{LP} \end{bmatrix} \tag{3}$$

where $\mathbf{f}^i$ is a $2r+1$ dimensional column vector.

$\mathbf{F}(L)$ is called the gray value matrix of the line segment L. Besides its own gray level information, it also contains the gray level information in around area of the line segment. However, the dimension of matrix $\mathbf{F}(L)$ will be changed with the different length of line segment. In order to get a gray value descriptor independent to the length of line segment, the mean value for each row of matrix $\mathbf{F}(L)$ is calculated, that is, $f_i^m = \frac{1}{LP}\sum_{j=1}^{LP} f_i^j$. And these mean values consist of a new $2r+1$ dimensional vector $\mathbf{F}_m(L) = \begin{bmatrix} f_1^m & f_2^m & \cdots & f_{2r+1}^m \end{bmatrix}$.

In order to make the line segments in different positions have different contribution to describe the feature line segment, a Gaussian weighting function is used to assign weight to each line in the line segment region. The Gaussian function is

$$W(L) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{d^2(L,l)}{2\sigma^2}\right) \tag{4}$$

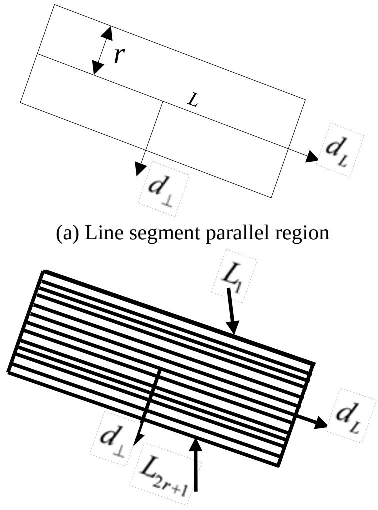
where $\sigma = r/2$, and $d(L,l)$ is the vertical distance of the line segment l to the feature line segment L.

So the line segment descriptor of weighted mean gray value is:

$$\mathbf{WF}_m(L) = \begin{bmatrix} W(l_1)f_1^m & W(l_2)f_2^m & \cdots & W(l_{2r+1})f_{2r+1}^m \end{bmatrix}$$)

Considered the impact of light, $\mathbf{WF}_m(L)$ need to be normalized as:

$$\mathbf{WMF}_m(L) = \frac{\mathbf{WF}_m(L)}{\|\mathbf{WF}_m(L)\|} = \begin{bmatrix} W(l_1)Mf_1^m & W(l_2)Mf_2^m & \cdots & W(l_{2r+1})Mf_{2r+1}^m \end{bmatrix} \tag{6}$$



(a) Line segment parallel region

(b) Parallel region decomposition

Figure 3. Line segment parallel region and its decomposition.

B. *Line segment descriptor of geometrical feature*

As shown in Fig. 4, M is the midpoint of line segment AB, θ is the angle between AB and x-axis, and α is the angle between OM and x-axis. In order to describe the geometric and spatial location properties accurately, the following four characteristic values for a line segment were defined.

Definition 1: $v_1 = \overline{AB}$, the length of line segment AB. $\overline{AB} = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$, where (x_1, y_1) and (x_2, y_2) are two endpoints of the line segment AB.

Definition 2: $v_2 = \theta$ is the directional angle of the line segment AB, where $\theta \in [0, \pi)$.

Definition 3: $v_3 = \alpha$ is the angle between OM to x-axis, where $\alpha \in [0, \pi/2]$.

Definition 4: $v_4 = \overline{MC}$ is the altitude of the midpoint M.

$\mathbf{V} = (v_1 \quad v_2 \quad v_3 \quad v_4)$ forms a geometrical feature descriptor, which describes the geometric and spatial location properties of a line segment.

In order to facilitate the selection of threshold and let the algorithm robust to scale variability, it is necessary to normalize the descriptor of geometrical feature. The normalized feature is represented as $\mathbf{V}'$. If there are N line segments in an image, the geometrical features can be expressed as the following matrix:

$$\mathbf{V}_{N\times4} = \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ \vdots & \vdots & \vdots & \vdots \\ v_{N1} & v_{N2} & v_{N3} & v_{N4} \end{bmatrix}_{N\times4} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_N \end{bmatrix} \tag{7}$$

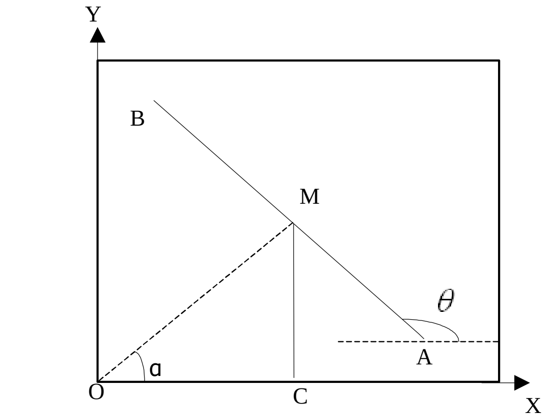


Figure 4. Geometrical features of line segment.

The matrix will be normalized as follows:

$$v_{i1}^{'} = v_{i1} \Big/ \sqrt{v_{11}^2 + v_{21}^2 + \cdots + v_{N1}^2}\ ,\ i = 1,2,\cdots,N \tag{8}$$

$$v_{i2}^{'} = v_{i2} / \pi\ ,\ i = 1,2,\cdots,N \tag{9}$$

$$v_{i3}^{'} = v_{i3} \Big/ \frac{\pi}{2}\ ,\ i = 1,2,\cdots,N \tag{10}$$

$$v_{i4}^{'} = v_{i4} \Big/ \sqrt{v_{14}^2 + v_{24}^2 + \cdots + v_{N4}^2}\ ,\ i = 1,2,\cdots,N \tag{11}$$

$$\mathbf{V}_{N\times 4}^{'} = \begin{bmatrix} v_{11}^{'} & v_{12}^{'} & v_{13}^{'} & v_{14}^{'} \\ v_{21}^{'} & v_{22}^{'} & v_{23}^{'} & v_{24}^{'} \\ \vdots & \vdots & \vdots & \vdots \\ v_{N1}^{'} & v_{N2}^{'} & v_{N3}^{'} & v_{N4}^{'} \end{bmatrix}_{N\times 4} = \begin{bmatrix} \mathbf{V}_1^{'} \\ \mathbf{V}_2^{'} \\ \vdots \\ \mathbf{V}_N^{'} \end{bmatrix}_{N\times 4} \tag{12}$$

IV. LINE SEGMENT MATCHING

A. *Epipolar-line constraint*

Using epopolar-line constraint can reduce the search range of matching point from the entire two-dimensional plane to a one-dimensional straight line. Thereby, the scope of the search range is reduced greatly, and the efficiency of point matching can be improved. For the line segment matching, epipolar-line constraint can also be used to improve the matching speed.

The principle of Epipolar-line constraint is shown in Fig. 5, two cameras are indicated by their centers C_1 and C_2. The line connecting two camera centers is called baseline. π_1 and π_2 are the image planes of left and right cameras, respectively. The baseline intersects two image planes at points e_1 and e_2, respectively. A point X in 3-D space is projected onto two image planes, at x_1 in the left, and x_2 in the right. The space point X and two camera centers are coplanar. The plane is denoted as epipolar-plane π, which intersects with the image planes formed corresponding epiploar lines l_1 and l_2. The point x_1 is correspond to epipolar-line l_2, and x_2 is correspond to epipolar-line l_1.Thus, there is a map $x_1 \to l_2$, from a point in one image to its corresponding epipolar line in the other image. This map is called epipolar-line constraint relationship.

The nature of the epipolar-line constraint relationship will be explored in this section. Each line segment has two endpoints, which have corresponding epipolar line in the other image. If the line segment is extracted integrally, the endpoints of matched line segment should be in the corresponding epipolar line. Generally, line segments may be extracted incompletely due to the influence of occlusion, illumination and other factors. The endpoints may not on their epipolar line, and matching line segments may not in corresponding region, which is determined by epipolar lines. So, the line segment can be seen as the candidate matching line segment [13] if a certain part of it located in the region. As shown in Fig.6 (a), point e is an epipolar of the left camera. x_1 and x_2 are endpoints of line segment L. In Fig.6 (b), the region determined by epipolar lines l_1 and l_2 is called matching region. The whole or part of two solid lines are in the matching region, namely, those lines are the candidate matching lines. The two dotted lines do not intersect with the matching region, namely, the non-matching line. So the usage of the epipolar-line constraint can greatly narrow the search area, and improve the matching speed.

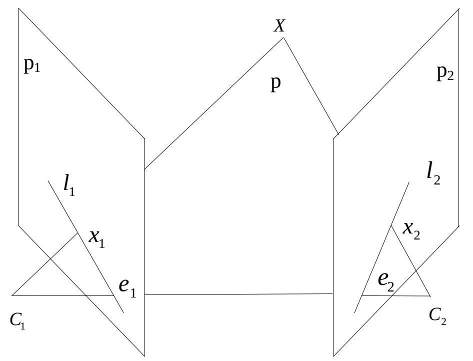
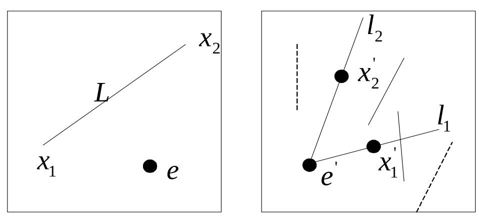


Figure 5. Epipolar-line constraint theory.



(a) (b)

Figure 6. The candidate matching region.

B. Line segments coarse matching

Geometrical feature of line segment describes the geometric and spatial location properties. When the change between two visual angles of left and right images is small, the change of the four characteristic values v_1, v_2, v_3 and v_4 of two matched line segments is small too. Therefore, there are following matching criteria. Let the ith line segment in the left image be denoted as $U_i^{'}$, there is N line segments in the left image, where $0 < i \le N$. Let the jth line segment in the epipolar-line constraint region of line segment $U_i^{'}$ be denoted as $V_j^{'}$, there is M line segments in this region and $0 < j \le M$. Use $u_{jk}^{'}$ and $v_{ik}^{'}$ to denote four features of normalized geometric descriptor of line segments $U_i^{'}$ and $V_j^{'}$, respectively, where $k = 1, 2, 3, 4$. There is:

$$\Delta v_{ik} = \left| v_{ik}^{'} - u_{jk}^{'} \right| < \varepsilon_k \tag{13}$$

where ε_k is a specific threshold. If the difference between two visual angles of left and right images is φ, then

$$\varepsilon_k = \frac{1}{2}\left(u_{ik}^{'} + v_{jk}^{'} \right) \cdot tg\varphi \tag{14}$$

When all values of Δv_{ik} are less than corresponding threshold, the two line segments are determined as the initial match pair. Using above method to all line segments in the epipolar-line constraint region of line segment $U_i^{'}$. Finding out all line segments that initial matched with $U_i^{'}$. Through the coarse matching process, the computation cost of the following line segment matching step can be reduced.

C. Line segments fine matching

Line segments are coarse matched based on the epipolar-line constraint and the matching rules of geometric descriptor. But there may be error matching of line segments for the coarse matching, a fine matching process is necessary. Firstly, normalized gray value descriptor $\mathbf{WMF}(L)$ and normalized geometric descriptor $\mathbf{V}_L^{'}$ are combined to a $2r+5$ dimensional vector $\mathbf{WMFV}(L)$, then $\mathbf{WMFV}(L)$ is use to describe line segment L.

$$\begin{aligned}\mathbf{WMFV}(L) = [&W(l_1)Mf_1^m \quad W(l_2)Mf_2^m \\ &\cdots \quad W(l_{2r+1})Mf_{2r+1}^m \quad v_{L1}^{'} \quad v_{L2}^{'} \quad v_{L3}^{'} \quad v_{L4}^{'}] \\ = [&wmfv_1 \quad wmfv_2 \quad \cdots \quad wmfv_{2r+5}]\end{aligned} \tag{15}$$

Secondly, the Euclidean distance between vectors is used to fine match of line segments. The Euclidean distance between line segments L and $L^{'}$ as follows:

$$DIS(L, L^{'}) = \sqrt{\sum_{i=1}^{2r+5} (wmfv_i - wmfv_i^{'})^2} \tag{16}$$

When $DIS(L, L^{'}) < \varepsilon$, the line segment L and $L^{'}$ is meet the condition of fine match, and there are matched each other. Where ε is a specific threshold.

When carrying out line segments fine match, only the Euclidean distance of line segments that meet the condition of coarse match need calculating.

V. EXPERIMENT RESULTS AND DISCUSSIONS

Some experiments are arranged to evaluate the accuracy and speed of the new algorithm. Both synthetic and real images are used in the experiments.

A. Synthetic images experiment

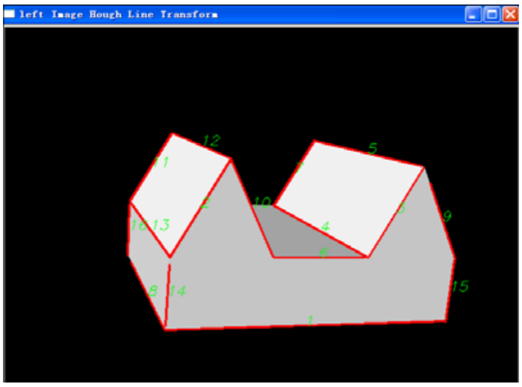
To verify the effectiveness of the algorithm, a set of synthetic images are selected firstly. The resolution of the synthetic image is 640×480. Line segments extracted for synthetic images were shown in Fig.7, both there are 16 line segments are detected in the right and left images. Use $r = 10$ to calculate the line segment descriptor $G_{10}(L)$ of mean gray value. The line segments matching results were shown in Fig.8, where the same serial number in the left and right images means a matched line pair, and there are 16 line segments have been matched totally. It is matched correctly 100% for the synthetic images without noise interference.

B. Real images experiment

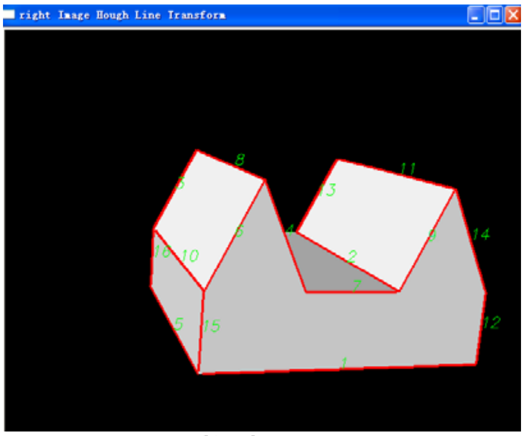
In this section, we will test the performance of matching algorithm on real images. Two images of a teaching model are shown in Fig.9, with a resolution of 615×461. As shown in Fig.10, both there are 16 line segments detected in the right and left images. Use $r = 10$ to calculate the line segment descriptor $G_{10}(L)$ of mean gray value, 16 line segments have been correctly matched as shown in Fig.11. More experimental results on real images are illustrated in Fig.12 and Fig.13. Images in Fig.12 are provided by Oxford University Visual Group. The line segments detection results are shown in Fig.13. There are 172 line segments detected in the right image and 174 line segments in the left image. Taken $r = 20$, the line segments matching results were shown in Fig.14. Total of 126 line segments are achieved matching, and number of 110 pairs line segments are correct matching. The result of algorithm in [8] compared with performance analysis of the algorithm of we propose in this paper, as given in Table I.

TABLE I. PERFORMANCE COMPARISON OF ALGORITHMS

Name	Which algorithm is used	The elapsed times	Reliability
Laboratory	The algorithm in[8]	1820ms	75%
Porch	The algorithm in[8]	1380ms	85.7%
Teaching model	The algorithm in this paper	59ms	100%
House	The algorithm in this paper	405ms	88%



(a) Left image



(b) right image

Figure7. Line segments extracted for synthetic images

C. *Results discussions*

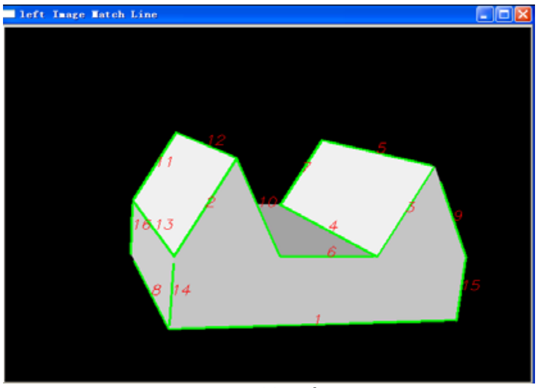
For the synthetic images and real images of simple teaching model, due to the line segments detection results are desirableness, all of the line segments are matched correctly. For complex real images, the correct matching ratio is about 88%. In addition, the coarse-to-fine matching processes avoid calculating Euclidean distance for all line segments effectively. That is, the line segments are matched coarsely based on epipolar-line constraint and geometric descriptor firstly; then matched finely based on Euclidean distance between vectors, which are $2r+5$ dimensional vectors combined geometric descriptor and gray value descriptor.

VI. CONCLUSION

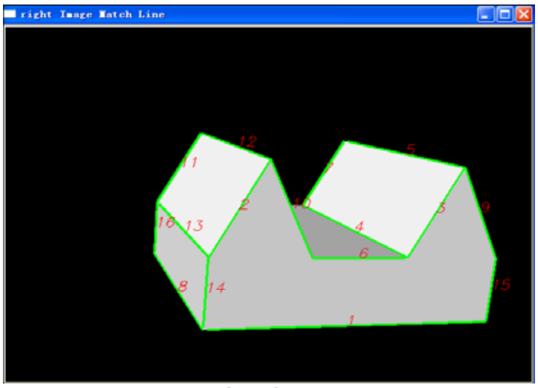
A line segments matching method is proposed in this paper. Compared with traditional methods, the proposed algorithm does not need a reference model, so the disability that can only matching line segments for a given model situation is avoided. The coarse-to-fine matching processes avoid calculating Euclidean distance for all line segments effectively. That is, the line segments are matched coarsely based on epipolar-line constraint and geometric descriptor firstly; then matched finely based on Euclidean distance between vectors. The Gaussian weighting function is used to assign weight to each line in the line segment region, avoid determining weighting factors for each line features manually.

Experiments show that the proposed algorithm performs line segments matching in a fast, robust, and efficient way. Moreover, the algorithm is scale invariable to some extent for the descriptors are normalized when describe a line segment. However, the threshold need setting manually when perform the fine line segments matching.

In order to improve the performance of the proposed line segment matching technique, further research will be focused on the reliability of real images.

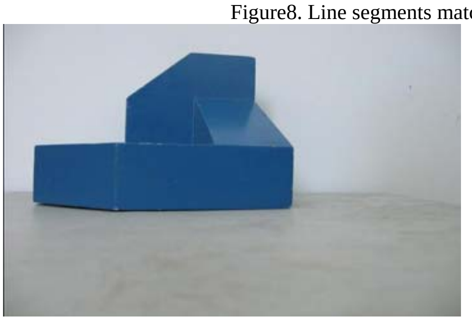


(a) Left image



(b) right image

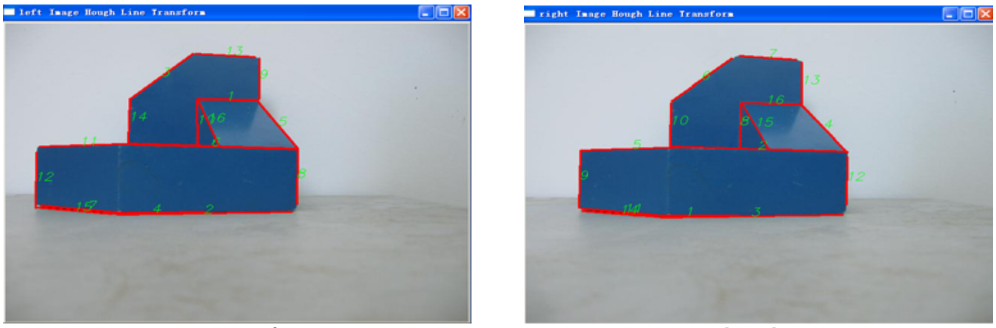
Figure8. Line segments matching for synthetic images.



(a) Left image

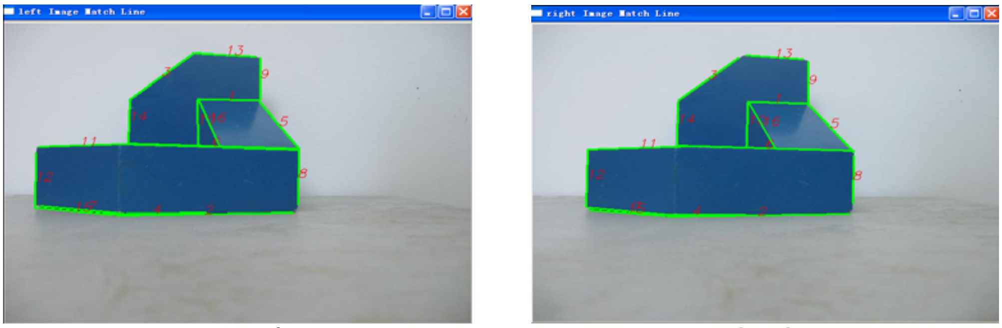
(b) right image

Figure9. Images of teaching model.



(a) Left image (b) right image

Figure10. Line segments extracted for the teaching model.



(a) Left image (b) right image

Figure11 Line segments matching for the teaching model.

Figure12. Image of the houses.

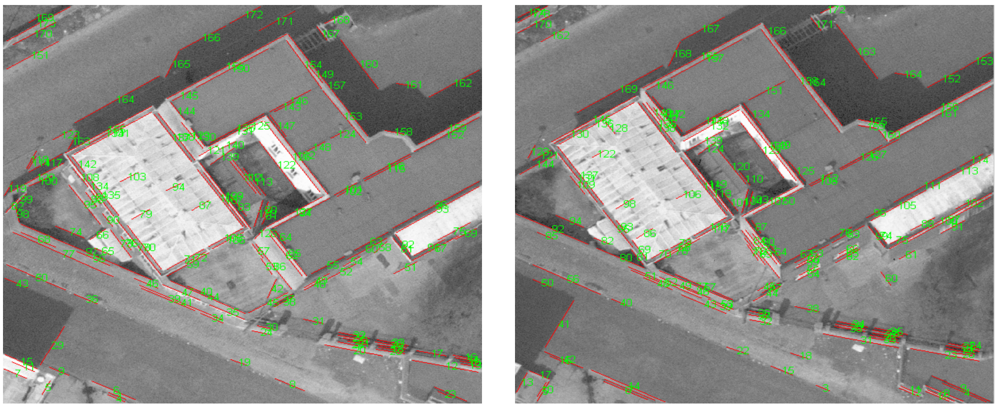


Figure13. The line segments extraction.

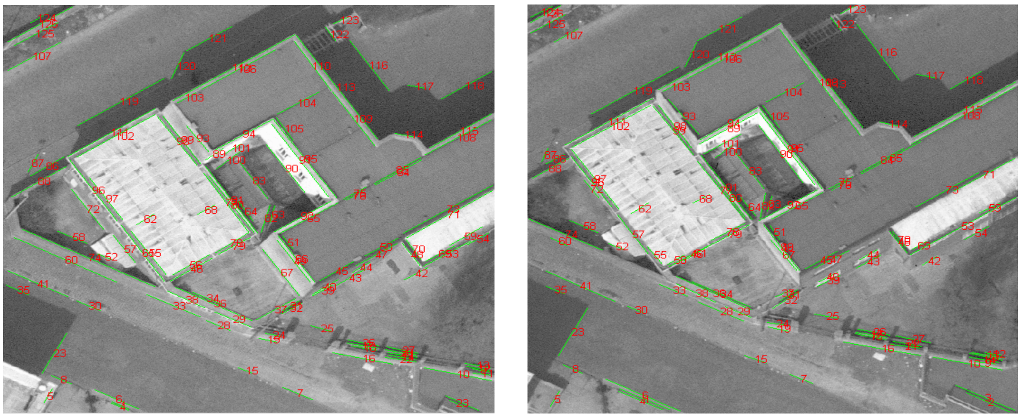


Figure14. The line segments matching.

ACKNOWLEDGMENT

This work was supported partially by the Aviation Science Foundation of China & National Key Laboratory of Science and Technology on Avionics System Integration (Grant No.20085556017).

REFERENCES

[1] J. S. Park, "Interactive 3D reconstruction from multiple images: a primitive-based approach," Pattern Recognition, 2005, vol. 16, pp. 2558-2571.

[2] K. P. Wang, Y. Sang and Q. F. Yu, "An Image Matching Approach Based on the Invariant Moment of Slope Angle Histogram," Journal of Computer Aided Design & Computer Graphics, 2009, vol. 21, pp. 388-93.

[3] G. J. Wen and R. S. Wang, "A global algorithm for straight line stereo matching based on feature grouping," Journal of Software, 2001, vol. 11, pp. 1660-1667.

[4] H. P. Sang, M. L. Kyoung and U. L. Sang, "A line feature matching technique based on an eigenvector approach," Computer Vision and Image Understanding, 2000, vol. 77, pp. 263-283.

[5] J. Ji, Ang Haisong etal., "Fast extraction and matching of straight lines on sequential images," Journal of Nanjing University of Aeronautics & Astronautics, 2005, vol. 37, pp. 227-231.

[6] M.I.A. Lourakis, S.T. Halkidis, S.C. Orphanoudakis, "Matching disparate views of planar surfaces using projective invariants," Image Vision Comput. 2000, vol.18, pp. 673–683.

[7] B. Herbert, F. Vittorio, V.G. Luc, "Wide-baseline stereo matching with line segments," IEEE International Conference on Computer Vision and Pattern Recognition, 2005, pp. 329-336.

[8] N. J. Chen and J. G. Li, "Match straight line based on line support region," Infrared and Laser Engineering, 2000, vol. 29, pp. 10-14.

[9] T. Li and X. L. Liu, "A robust approach for extracting and matching straight line," Computer Simulation, 2008, vol. 25, pp. 171- 173.

[10] Y. Deng, X.Y. Lin, "A fast line segment based dense stereo algorithm using tree dynamic programming," European Conference on Computer Vision, 2006, pp. 201-212.

[11] Z. H. Wang and F. C. Wu, "Mean Standard Deviation Descriptor and Line Matching," PR&AI, 2009, vol. 22, pp. 32-39.

[12] J. Cha and R. H. Cofer, "Extended hough Transform for linear feature detection," Pattern Recognition, 2006 vol. 39, pp. 1034-1043.

[13] C. Baillard ,C. Schmid and A. Zisserman, "Automatic line matching and 3D reconstruction of buildings from multiple views," In ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, UK, 1999, pp. 69-80.

Jie-xian Zeng was born in Le-an, Jiangxi province, China in 1958. He obtained a Master of Science (M.S.E.) degree in engineering graphics from Northwestern Polytechnical University, Xi-an, China in 1997. He is currently a professor of Nanchang Hangkong University China, and has published more than 50 papers. His research interests include computer graphics, theoretical graphics, computer vision, and pattern recognition. He can be reached by e-mail: zengjx58@163.com or by telephone: 086-0791-3953403.

Sheng Yu, he is born in Hunan Loudi China, and born in 1982, 11. He received his MS degree from Nan Chang Hang Kong University of China in 2010. He has been engaged in research in image processing and pattern recognition. He can be reached by e-mail: yu_sheng4105@163.com.

Xiang Fu was born He-feng, Hubei, China in 1980. He obtained a Master of Science (M.S.E.) degree in computer application and Ph.D degree in Circuits and Systems from Xidian University, Xi-an, China respectively in 2005 and 2008. He is currently an instructor of Nanchang Hangkong University, Nanchang, China. His research interests include image processing and pattern recognition. He can be reached by e-mail: fxfb163@163.com.

Chao-xia Li. She is born in Shanghai China, and born in 1975,11. She received his MS degree from Northwestern Polytechnical University of China in 2000. She is currently an researcher of National Key Laboratory of Science and Technology on Avionics System Integration, Shanghai, China, where she has been engaged in research in image processing and analysis, and pattern recognition. She can be reached by e-mail: lizhaoxia2002@yahoo.com.cn or by telephone: 086-13301932162.