Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 13 (18): 3736-3741, 2013
ISSN 1812-5654 / DOI: 10.3923/jas.2013.3736.3741

A Fast Three-Phase Line Segments Clustering Method Based on Relative Spatial Relationship

[1,2,3]Y.Q. Liu, [2]X.H. Su and [2,3]E.H. Wu
[1]School of Information Engineering, Chang'an University, Xi'an, 710064, China
[2]State Key Lab of Computer Science, Institute of Software, CAS, Beijing, 100190, China
[3]Faculty of Science and Technology, University of Macau, Macao, China

Abstract: Lines indicate structure information of objects. However, the general line detectors cannot give enough clear information with many short or discontinuous line segments. This study presents a new fast three-phase line segment clustering algorithm. Firstly, Hough transform or LSD algorithm is used to attain initial line set; and then these lines are grouped into different sets according to direction; and then each direction set is further subdivided into different sub-sets according to their relative distances; finally the lines are merged or split on the basis of their neighborhood relations to form the final groups. Compared to previous work, the present method is more efficient and easier to implement. More importantly, the clustered line segments can fully indicate the structure information of targets in the image which is verified by the experiments.

Key words: Line segment clustering, structure feature, distance, direction, merging or splitting

INTRODUCTION

Lines indicate important structure information of images; they give the global information of target instead of local illumination change. So, line detection is very important in computer vision and image processing community. However, the detected lines from the usual image processing techniques are often at a mess and discontinuous, making it hard to identify the structure of the object. Therefore a cluster algorithm is needed to reorganize these mess lines to discriminate the object structure which serves as the input for further vision tracking and identification.

For general data clustering, k-means algorithm is often used which aims at point data clustering. Gong *et al*. (2011) presented a two-phase method for complex distributed data, even though they could cluster the complex data into different groups, they started from sphere-like distribution. However line segments have strong direction property, they are high-dimensional data as, where $\vec{p}_c$ is the center point of the line segment, θ is the direction and l is the length. Thus the general data clustering algorithms cannot satisfy the requirement for lines clustering.

As to line clustering, Kostas *et al*. (2004) presented a scan-line grouping algorithm which connect short straight line segments into long curve. However this work only scans from the specified direction which limits the range of application. Bazin *et al*. (2012) used parallel lines to evaluate vanish point from Manhattan images. For general line segments, Nacken (1993) proposed a line metric to merge two neighbor line segments. Park (2010) proposed a hierarchy clustering method to exact the rectangle boundary from aerial images to detect buildings which used centroid neural network to connect low-level linear structure or group similar objects. But it is limited to rectangle shapes. Jonk and Smeulders (1995) gave a scale-invariant hierarchy method to merge close short line segments, but both trigonometric function and neighborhood function are used which are very time consuming.

Jang and Hong (2002) presented a fast line segments grouping method by using direction metric and distance metric, this method overcome the limitation of traditional methods. But in each step the parameters are updated by solving an eigenvalue problem which is very time consuming. Inspired by this work, a new novel three-phase line segments clustering method is given which uses step-by-step strategy to divide the short broken line segments into different groups with merging or splitting used.

METHODS

As the input, the initial line segments are firstly detected with a line detector from image processing. This

Corresponding Author: Y.Q. Liu, School of Information Engineering, Chang'an University, Xi'an, 710064, China

study gets the initial lines from Hough (1962) transformation with Canny (1986) operator, or LSD algorithm by Von Gioi *et al*. (2010).

Since the line segment is determined by, here gives the position information of the line segment, θ gives the direction information of the line and l gives the geometry information of the line segment. From the point of view of clustering, the line segments in different directions obviously belong to different groups even though they are close or similar long. But for a group of lines which are in the same direction, the distance metric begins to play an important role to identify themselves. Those close line segments are more likely in the same group. From the vision cognition, the length of line segments illustrates the significance level of the structure feature, that's, longer line segments are more obvious and more important. Current researches tried to combine these three parameters into a single formula which seems direct and compact, but failed to indicate the physical meanings with more complexity.

Thus as shown in Fig. 1, a three-phase method is designed to cluster the lines into suitable groups according to the intension of the three parameters.

Rough direction clustering: Different from point data, line segments are directional vector data, different direction will classify these line segments into different group naturally, that's, here.

As to the line segment L_i, actually use is used instead of to represent it by replacing θ with direction vector $\vec{N}$ to avoid complex trigonometric function calculation. During implementation, each input line segment's direction is compared with the direction of each element in the available direction group. If the angle between the new line segment L_i and one sub-group is smaller than a threshold, then L_i is added into. Otherwise, a new

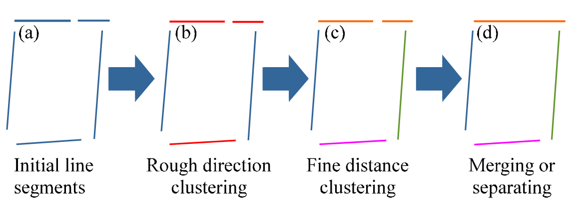

Fig. 1(a-d): The procedure of the present algorithm which is a three-phase clustering method with (b), (c), (d) steps in sequenc,. (a) Input initial line segments attained from original image, (b) Result from rough direction clustering, (c) Result from fine distance clustering and (d) Final clustering result after merging or splitting operation

direction sub-group is created with L_i added as the first element. At the same time is added into as an element. Jang and Hong (2002) used the least square method to calculate the accumulator matrix to update the direction information of each sub-group which was complex and time-consuming. By contrast, in this study since the length indicates the important degree, the direction of the longest line segment is selected as the direction of the whole sub-group.

The waiting line segments will compare the direction with by dot product to determine whether to join the sub-group or not. Different to Jang and Hong (2002) work, this algorithm is more efficient with the pseudo-code shown as follows in Algorithm 1.

Algorithm 1: Pseudo-code of rough direction clustering

Input: {L},ε
Output: {D}
1. foreach line L_i in {L} do
2. notfound←true;
3. foreach $\{D_j\} \in \{D\}$ do
4. $dot \leftarrow \vec{N}_{\{D_j\}} \cdot \vec{N}_{L_i}$;
5. if
6. notfound←false;
7. push back L_i into $\{D_j\}$;
8. update $\vec{N}_{\{D_j\}}$;
9. break;
10. end
11. if notfound
12. add a new group $\{D_x\}$ into {D};
13. push back L_i into $\{D_x\}$;
14. set $\vec{N}_{\{D_x\}} \leftarrow \vec{N}_{L_i}$;
15. end

Further fine distance clustering: With the above rough direction clustering, all line segments are classified into different groups. But for each group, the difference among these line segments is still obvious, too close or too far. So the direction result is further clustered according to the distance in each direction. Since the distance between two segments is so diverse, the shortest distance between the two line segments can not be used as the criterion for clustering. For example, two line segments are on the same base line, just because of the large distance between the end points, they will be wrongly partitioned into different groups. This study proposed a simple and effective method to avoid such kind of errors. The distance from the center point $\vec{p}_c$ of the line segment for grouping to the base line of the available groups is used as the cluster. Since distance clustering is done based on the direction groups which means that for each sub-set in, it is refined to construct the distance set. Similar to direction grouping, the longest line segment is selected as the representive of sub-set to serve as the base line, not line segment. The detail procedure of this step is in Algorithm 2.

Algorithm 2: Pseudocode of further fine distance clustering

```
Input: {D},ε
Output:{S}
1. foreach subset {D_j}∈{D} do
2. notfound←true;
3. foreach L_i in D_j do
4. foreach subset {S_k}∈{S} and {S_k}⊂{D_j} do
5. calculate distance between L_{S_k} and P_{c L_i}
6. if dis<ε
7. notfound←false;
8. push back L_i into S_k;
9. update L_{S_k} ;
10. break;
11. end
12. if notfound
13. add a new group {S_k} into {S};
14. push back L_i into {S_k};
15. set L_{S_k} ← L_i ;
16. end
17. end
```

Merging or splitting: With the two aforementioned clustering steps, all line segments are grouped into different sets with direction and distance discriminated. However, there are many short line segments with much noise. Some long line segments are separated into several parts due to noise or the weakness of line detector algorithms. So it is necessary to remedy the cluster result. Figure 2, in some cases the line segments belong to the same group, but they are divided into small short segments. This mainly comes from the inefficiency of line detectors which cannot determine the whole line just based on gray gradient. In this study the cluster result is fixed to reflect the correct structure information, as shown in Fig. 2a-b merging used, while in Fig. 2c separating used to discriminate the far apart objects.

Different from clustering procedure, merging or splitting step will re-organize the line segments with the structure information modified. Based on clustering result

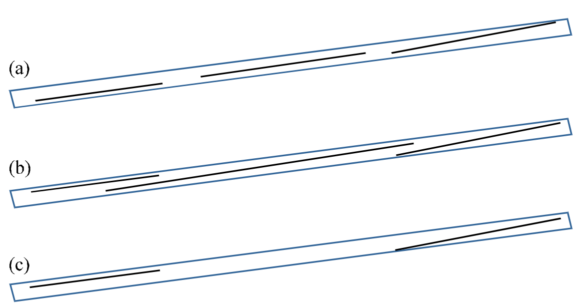

Fig. 2(a-c): Three different overlapping situations of the line segments in the same direction with (a), (b), (c), (a) Gives a example with short discrete line segments; (b) Gives a example with overlapping line segments and (c) Gives a example with too far line segments

{S}, the average direction $\vec{N}_{average}$ and the center point P of each sub-set $\{S_k\}$ is computed. To decrease the influence of the noise line segments, the length of line segment is used as the weight function. All line segments in are projected onto the direction to calculate the range parameter around P. Then is compared with of each sub-set in. If or, the processing line segment is far from other line segments in, it should belong to other available set or a new set; otherwise this line segment overlaps with the current sub-set, so is used to update to integrate this line segment into the sub-set. Note that during the merging procedure, all line segments in $\{S_k\}$ will be sorted in $t_{min, L}$ so that the line segments are merged from left to right without missing any line segments. And **ε** is 20 pixels in this study. The algorithm of this step is illustrated in Algorithm 3.

Algorithm 3: Pseudocode of merging or splitting

```
Input: {S},ε
Output: {F}
1.foreach subset {S_j}∈{S} do
2. notfound←true;
3. calculate N_average of {S_j};
4. calculate P_center of {S_j};
5. foreach L_i in {S_j} do
6. calculate [t_min,L, t_max,L] around P_center , N_average ;
7. end
8. sort t_min,L from left to right
9. foreach L_i from left to right in S_j do
10. foreach subset do
11.    if
12.     continue;
13.    else
14.     notfound←false;
15.     push back L_i into {F_k};
16.     update with;
17.     break;
18. end
19. if notfound
20.    add a new group {F_x} into {F};
21.    push back L_i into F_x;
22.    set ;
23. end
24.end
```

RESULTS AND DISCUSSION

The algorithm presented in this study is implemented with C++. The hardware platform is equipped with Intel Core I3 CPU@2.13GHz, 2G memory, OS is Win7. To verify the effectiveness of the algorithm a group of test data is designed as shown in Fig. 3(a-f) with some noise or short line segments added. Readers can visit project website http://imlab.chd.edu.cn/line/ to find the original size images.

To verify the efficiency of the present algorithm, it is compared with the method of Jang and Hong (2002) as shown in Table 1. From Table 1, it is clear that the method of this study is more efficient with the similar results

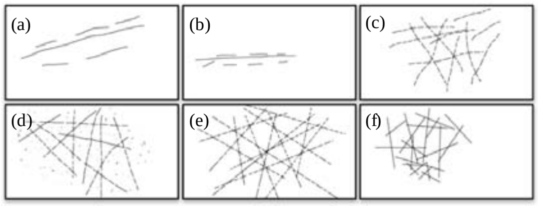

Fig. 3(a-e): Six original test pictures using as the input images in the experiments. (a), (b), …, (e) give six examples with different line distribution, respectively

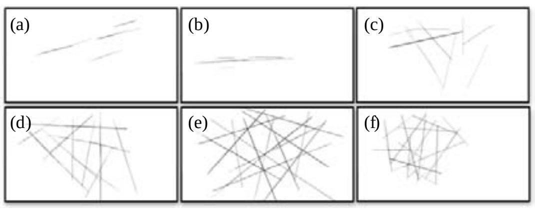

Fig. 4(a-e): The corresponding final cluster results of the according test pictures in Fig. 3 with the present algorithm. (a), (b), …, (e) are the results of the according (a), (b), …, (e) in Fig. 3

Table 1: Performance comparison between the present algorithm and Jang's algorithm which is illustrated with time cost in second

Test image	The cost of present algorithm (sec)	The cost of Jang *et al.*'s method (sec)	Lines detected
Fig. 3a	0.000083	0.000098	4
Fig. 3b	0.000057	0.000085	3
Fig. 3c	0.000120	0.000560	8
Fig. 3d	0.000242	0.000618	11
Fig. 3e	0.000322	0.000959	15
Fig. 3f	0.000322	0.000472	20

shown in Fig. 4, even in some case the performance is three times faster than the method of Jang and Hong (2002). For some missing line segments not shown, the reason is that the initial line detector does not detect them successfully as shown in Fig. 5.

For vision tracking or aerial image identification, good line segments method will provide clear object structure information. So some real image data including interior and outdoor scenes are tested.

In Figure 6, the doorframe in the final result is much clearer than the result from line detector without too much noise. The structure of the table is much clear in final result of Fig. 7a, as well as the house structure in Fig. 7b. In Fig. 7c, the fence structure can be identified. From Fig. 6-7, it is clearly shown that the present algorithm can capture the structure information very well through line segments clustering

However, there are still some limitations in this study. If the initial line segment set does not have enough lines to represent the structure of objects, then the cluster algorithm won't work fine to indicate the object. For artificial data, Hough transformation can give a good

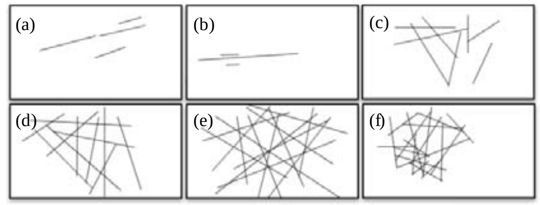

Fig. 5(a-e): The corresponding initial line segment set from the according test pictures in Fig. 3 which are used as the input line sets. (a), (b), …, (e) are the results of the according (a), (b), …, (e) in Fig. 3

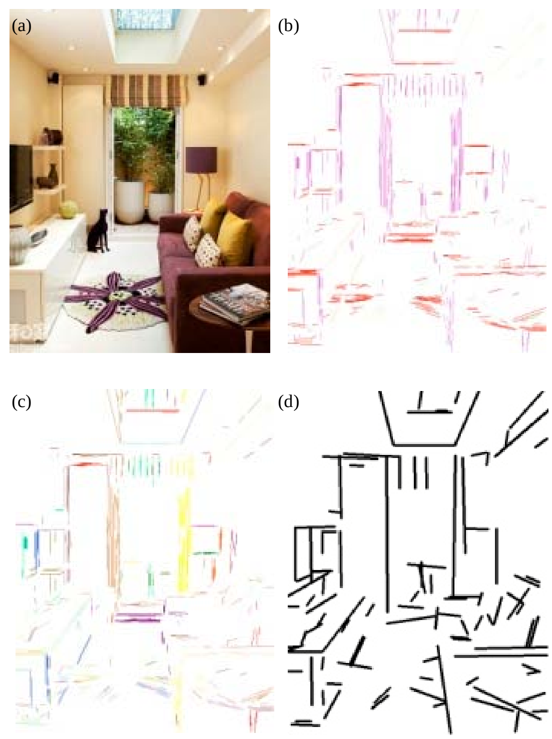

Fig. 6(a-d): Experiment procedure with interior scene image used. (a) Input original image; (b) Direction clustering result from (a); (c) Distance cluster result from (b); (d) Final clustering result from (c)

Fig. 7: Experiment results of several test image about real scene, (a), (b) and (c) give three examples to illustrate the right-column final clustered line segments from the left-column original image, respectively

initial line segment set while the LSD algorithm has the immense advantage in natural scenes. However, based on a good initial line segment set, the algorithm present here can reflect object structure feature very well.

CONCLUSION

This study presents a novel fast line segment algorithm which consists of rough clustering from direction, fine clustering from direction, merging or splitting. Compared to previous work, the algorithm is more efficient and easier to implement. Such kind of clustering can provide much clearer structure of objects than the single line detecting without too much junk information to interfere the structure which help improve vision tracking and recognition.

However, some limitations can be improved in the future, like the results depending on the initial detected line segments. If the initial line segments are not ideal, the clustering result will fail to represent the structure clearly. Another limitation is that the single straight line segment clustering has restricted expression capability. Curve line clustering will be investigated to embody the structure information from images.

ACKNOWLEGMENTS

This study is financially supported by NSFC (60973066), open fund of the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences (SYSKF1004), the Special Fund for Basic Scientific Research of Central Colleges (CHD2011TD009, 2013G1241112, 2013G3242008), Program for Changjiang Scholars and Innovative Research Team in University (IRT0951).

REFERENCES

Bazin, J.C., Y. Seo, C. Demonceaux, P. Vasseur, K. Ikeuchi, I. Kweon and M. Pollefeys, 2012. Globally optimal line clustering and vanishing point estimation in manhattan world. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 16-21, 2012, Providence, RI., USA., pp: 638-645.

Canny, J., 1986. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell., PAMI-8: 679-698.

Gong, M.G., S. Wang, M. Ma, Y. Cao, L.C. Jiao and W.P. Ma, 2011. Two-phase clustering algorithm for complex distributed data. J. Software, 22: 2760-2772.

Hough, P.V.C., 1962. A method and means for recognizing complex patterns. US Patent Number, 3069654.

Jang, J.H. and K.S. Hong, 2002. Fast line segment grouping method for finding globally more favorable line segments. Pattern Recogn., 35: 2235-2247.

Jonk, A. and A.W. Smeulders, 1995. An axiomatic approach to clustering line-segments. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Volume 1, August 14-16, 1995, Montreal, Canada, pp: 386-389.

Kostas, K.V., A.I. Ginnis and P.D. Kaklis, 2004. A scan-line algorithm for clustering line segments. Proceedings of the Conference on Shape Modeling Applications, June 7-9, 2004, Genova, Italy, pp: 379-392.

Nacken, P.F.M., 1993. A metric for line segments. IEEE Trans. Pattern Anal. Mach. Intell., 15: 1312-1318.

Park, D.C., 2010. Hierarchical clustering of 3-D line segments for building detection. Proceedings of the IEEE International Symposium on Signal Processing and Information Technology, December 15-18, 2010, Luxor, Egypt, pp: 388-393.

Von Gioi, R.G., J. Jakubowicz, J.M. Morel and G. Randall, 2010. LSD: A fast line segment detector with a false detection control. IEEE Trans. Pattern Anal. Mach. Intell., 32: 722-732.