ELSEVIER

Pattern Recognition Letters 17 (1996) 643-649

Pattern Recognition Letters

Extracting straight lines by sequential fuzzy clustering

Koji Tsuda *, Michihiko Minoh, Katsuo Ikeda

Department of Information Science, Kyoto University, Kyoto 606-01, Japan

Received 22 September 1995; revised 29 November 1995

Abstract

In clustering line segments into a straight line, threshold-based methods such as hierarchical clustering are often used. The line segments comprising a straight line often get misaligned due to noise. Threshold-based methods have difficulty clustering such line segments. A new cluster extraction method is proposed to cope with this problem. This method extracts fuzzy clusters one by one using matrix computation. We evaluated our clustering method using hand-written drawings and obtained promising results.

Keywords: Line extraction; Clustering; Rough sketch reforming

1. Introduction

Extraction of lines from an image is an essential task in computer vision. Many authors have reported various algorithms for line extraction, such as Hough Transformation and gradient-based methods (Burns et al., 1986). An effective approach to line extraction is by clustering line segments (Mohan and Nevetia, 1992; Nacken, 1993). Line segments are obtained by applying a line fitting process to the output of edge detection process. A line segment is a small geometric structure defined by two end points. In many cases, line segments lie along long straight lines. By clustering these segments, straight lines in the image are extracted and the large linear structure can be found.

This approach is applicable to reforming a rough sketch, which we are currently studying (Minoh et al., 1995). A rough sketch is a kind of record of the design process and so several ideas are presented at the same time in a sketch. A rough sketch has the characteristic that several lines are drawn to represent a line. Our primary objective is to cluster such lines into the line which the designer wanted to draw. This clustering can help the designer summarize his ideas. We call this task *rough sketch reforming*. Since this task is similar to the line extraction task in computer vision, line extraction algorithms can easily be applied to rough sketch reforming.

To gather line segments into a longer line, some grouping technique is needed. Such a technique usually consists of two independent phases. First, the "collinearity" (or similarity) of two line segments is measured based on their directions and on the distance between their end points. Second, the clustering of line segments is carried out. In our research, we will use the ordinal measure of collinearity and focus our attention on the second part – the clustering algorithm.

To summarize the ideas of the designer effectively, the structure of the things written in the sketch should be represented as simply as possible. So, the line segments should be clustered into a small number of

* Corresponding author. E-mail: koji@kuis.kyoto-u.ac.jp.

PII S0167-8655(96)00029-3

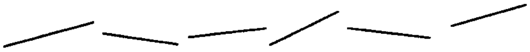


Fig. 1. An example of weakly-collinear line segments.

straight lines. Therefore, the number of the segments in each cluster should be large.

However, ordinary line segment clustering methods focus only on the collinearity between segments, and pay no attention to the number of the segments in each cluster, since these methods use a threshold of collinearity to produce clusters. These threshold-based methods have a major defect: a poor ability to cluster weakly-collinear line segments into a straight line.

The line segments that are derived from a straight line in the image are not necessarily oriented in the same direction. The directions of line segments deviate from the true direction of the original line during edge extraction, thinning and line fitting. We call such line segments weakly-collinear segments (Fig. 1). Since the collinearity among the segments is small due to the difference of directions, threshold-based methods often fail to cluster the segments into a straight line. Instead, the segments are likely to be divided into several small straight lines, which is undesirable for simplifying the sketch.

We propose a new clustering method named Sequential Fuzzy Cluster Extraction Method (SFCE). It is an optimization-based method: a criterion function that shows the goodness of a cluster is established, and a cluster is obtained by maximizing the function under specific conditions. Since this criterion function is determined as the sum of all collinearities among the segments in a cluster, the criterion function reflects the number of the segments in a cluster as well as their collinearities. Thus, if the number of weakly-collinear segments is sufficiently large, the segments can be clustered into a straight line by SFCE.

In Section 2, previous methods are reviewed and their problems are pointed out. In Section 3, the SFCE method is described as an optimization problem. In Section 4, we show that this optimization problem can analytically be solved by matrix computations. In Section 5, SFCE is applied to a rough sketch and a synthetic image for evaluation. Finally, we conclude our discussion in Section 6.

2. Previous methods

Let us review some of the previous line segment clustering methods. The two well-known methods are the thresholding method (Mohan and Nevetia, 1992) and the hierarchical clustering method (Nacken, 1993). In the thresholding method, two line segments whose collinearity is above a certain threshold are considered to belong to the same line. A cluster is formed by traversing the segments whose collinearities are above the threshold. The advantage of the method is its simplicity and speed. The time complexity of this algorithm is $O(n^2)$ and the storage requirement is $O(n)$, where n is the number of line segments.

In the hierarchical clustering method, the most collinear pairs of lines are repeatedly merged into a longer line. The output of this algorithm is a binary tree whose terminal nodes are the original line segments. Each non-terminal node is a line segment produced by the merging of its two children. Lines can be obtained by cutting this tree at a level where the collinearity between lines is less than a threshold. This is a very slow method; the time complexity of this algorithm is $O(n^3)$ and the storage requirement is $O(n)$. Its advantage over the thresholding method is that the threshold can be determined after grouping.

These two methods have one characteristic in common: they both use a threshold to obtain clusters. The main difference between them is the phase where thresholding takes place: before or after grouping. In these methods, the proper setting of the threshold is very important to obtain meaningful results. If the threshold is too high, the line segments will be divided into many clusters so that the reforming process is almost meaningless. On the other hand, if the threshold is too low, two kinds of problems will occur: *overclustering* and *noise inclusion*.

Since the sketch is to be simplified, the number of extracted straight lines should be small. But the effort to cluster the line segments into fewer straight lines often causes overclustering, merging two obviously distinct lines into one. Overclustering distorts the structure of the objects in the image and damages the visual impression of the reformed result.

The line segment fitting process produces many randomly-oriented short segments. We call these segments "noise segments" or simply "noise". For example, if the image contains textured areas (e.g. trees in scene images), these areas will be filled with noise. If noise segments are by chance collinear and the collinearity between them is above the threshold, they are clustered as a straight line. We call this phenomenon noise inclusion. Noise inclusion emphasizes trivial lines that do not belong to the structure of the things in the sketch, and confuses the reformed result.

Since the collinearity of the weakly-collinear segments is small due to the difference of directions, the threshold must be low to extract these segments as a line. However, a low threshold produces many over-clustered segments and noise inclusion. In some cases, it is impossible to cluster weakly-collinear line segments without these problems.

3. Sequential fuzzy cluster extraction

In SFCE, clusters are represented in a fuzzy fashion. Each object i ($1 \leqslant i \leqslant n$) has a membership value w_i ($0 \leqslant w_i \leqslant 1$) which represents how much an object i belongs to a cluster. To normalize membership values, the quadratic sum of all membership values is set to 1, that is

$$\sum_{i=1}^{n} w_i^2 = \|w\|^2 = 1, \tag{1}$$

where w is the n-dimensional vector of memberships. In our method, extraction of a cluster means determining membership values of all objects, which is achieved by solving an optimization problem. The reason why we choose this quadratic normalization scheme instead of a more obvious one such as $\sum_{i=1}^{n} w_i = 1$ is to easily solve the resultant optimization problem. This point will be described later in Section 4.

The criterion function of SFCE is designed to reflect the number of objects in a cluster as well as the similarity among them. So, the criterion function is determined as the sum of all similarities between members of a cluster. Let e_{ij} be the similarity value between object i and j ($e_{ij} = e_{ji}, e_{ij} \geqslant 0$), the criterion function $Q(w)$ is formulated as

$$Q(w) = \sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} e_{ij} w_i w_j. \tag{2}$$

The cluster extraction is formulated as the optimization problem to find an optimal solution w^* that maximizes $Q(w)$ under the constraint of Eq. (1).

We can obtain only one cluster by solving the optimization problem. To obtain more clusters, the optimization problem must be solved repeatedly. First of all, the first cluster w_1 is obtained as a solution of the following optimization problem: Maximize $Q(w_1)$ subject to the constraints

$$\sum_{i=1}^{n} w_{1i}^2 = 1, \quad w_{1i} \geqslant 0. \tag{3}$$

The members of the second cluster w_2 need to be extracted from the objects that do not belong to the first cluster. So, the duplication measure of the first and second cluster $dup(w_2, w_1)$ should be minimized, where $dup(w_k, w_l)$ is defined as follows:

$$dup(w_k, w_l) = \sum_{i=1}^{n} w_{ki}^2 w_{li}^2. \tag{4}$$

The duplication measure $dup(w_2, w_1)$ is added as a penalty term to the objective function. w_2 is obtained by solving the following problem: Maximize

$$Q(w_2) - \beta_1\, dup(w_2, w_1) \tag{5}$$

subject to the constraints

$$\sum_{i=1}^{n} w_{2i}^2 = 1, \quad w_{2i} \geqslant 0. \tag{6}$$

The kth cluster is obtained in a similar way. The penalty term is the weighted average of the duplications $dup(w_k, w_1), \ldots, dup(w_k, w_{k-1})$. w_k is obtained by solving the following problem: Maximize

$$Q(w_k) - \frac{1}{k-1} \sum_{t=1}^{k-1} \beta_t\, dup(w_k, w_t) \tag{7}$$

subject to the constraints

$$\sum_{i=1}^{n} w_{ki}^2 = 1, \quad w_{ki} \geqslant 0. \tag{8}$$

The parameter β_i (> 0) controls the degree of duplication of the ith cluster with the other clusters.

Since we do not have an established method for setting the β_i's, these parameters must be set empirically.

To ease the effort to set them, we assume every β_i is proportional to $Q(w_i)$, and reduce the number of parameters to one, that is

$$\beta_i = \eta Q(w_i), \tag{9}$$

where the parameter η represents the duplication ratio.

4. Matrix computation for extracting clusters

The optimization problem presented in the previous section can be converted into an eigenvalue problem of an $n \times n$ matrix by the following theorem.

Theorem 1 ((Bellman, 1970)). *The first eigenvector (i.e. the eigenvector with the largest eigenvalue) z_1 of an $n \times n$ symmetric matrix H is the global optimal solution of the following optimization problem: Maximize*

$$\sum_{i=1}^{n}\sum_{j=1}^{n} h_{ij}x_i x_j \tag{10}$$

subject to the constraints

$$\sum_{i=1}^{n} x_i^2 = 1. \tag{11}$$

Let w_k^* denote the global optimal solution of Eq. (7). If the elements of the matrix H are set as follows:

$$h_{ij} = \begin{cases} e_{ij} & (i \neq j), \\ -\dfrac{1}{k-1}\displaystyle\sum_{t=1}^{k-1} \beta_t w_{ti}^2 & (i = j), \end{cases} \tag{12}$$

then w_{ki}^* can be derived from the first eigenvector z_1 of H. That is,

$$w_{ki}^* = |z_{1i}|. \tag{13}$$

The derivation of Eq. (13) from Eq. (12) is given in Appendix A.

The optimization problem presented in Theorem 1 is a special case in which the global optimal solution can be analytically solved by matrix computations. So, if we do not normalize the membership values in a quadratic way as in Eq. (1) or do not define the criterion function as in Eq. (2), we must use a nonlinear optimization technique and due to many local optimal solutions, finding the global optimal solution is very difficult.

To calculate the first eigenvector, the Lanczos method is used (Cullum and Willoughby, 1985). The time complexity to obtain the first eigenvector is $O(n^2)$, and the storage requirement is $O(n)$. Since the eigenvector computation is needed for every cluster, the time complexity of SFCE is $O(cn^2)$, where c ($\leqslant n$) is the number of clusters. The time complexity is therefore between those of the thresholding method and hierarchical clustering.

5. Experiments in line extraction

5.1. Measure of collinearity

In the following experiments, a similarity value between two line segments is determined by the parameters shown in Fig. 2. This similarity value is

$$\frac{l(T_d - d/l)(T_s - s)(T_c - c)}{T_d T_s T_c}, \tag{14}$$

where l is the average of l_1 and l_2, the lengths of two segments, d is the distance between the nearest end points, s is the angle between the two segments, c is the average of c_1 and c_2, which are the angles between the line connecting the centers and the two line segments, T_d, T_s and T_c are constants that determine the effective ranges of the parameters. If the similarity value is less than zero, then it is set to zero.

This similarity measure is devised from that of (Minoh et al., 1995), where the measure of each feature (e.g. the distance between two end points) is independently defined. These measures are unified to be applicable for clustering methods.

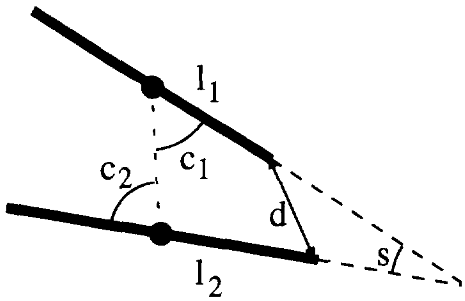


Fig. 2. Parameters of the measure of collinearity.

5.2. Replacing line segments by a single line

When clusters of line segments are obtained, each cluster has to be represented by a single straight line. Since clusters are represented in a fuzzy fashion in SFCE, we have to defuzzify the clusters first. A fuzzy cluster w_k is defuzzified to a hard cluster C_k according to the following procedure. Initially, C_k is a null set. The segments are added to C_k one by one in descending order of membership value w_{ki}. The adding process continues until

$$\sum_{i \in C_k} w_{ki}^2 > \sigma, \tag{15}$$

where σ is a parameter that controls the boundary of the cluster.

The straight line representing the segments in C_k is obtained as follows: First, the straight line that minimizes the mean squared error of all the end points of the segments to the line is constructed. Then, the segments are projected onto the line. The end points of the straight line are determined as the two most distant points among the end points of the projected segments. In hierarchical clustering, this replacement procedure is used to merge two segments in the course of growing clusters. In SFCE, this replacement procedure is used after all clusters are found. However, each cluster may be replaced one by one right after it is found, since the clustering procedure and the replacement procedure are independent.

5.3. Experimental results

Both SFCE and hierarchical clustering were applied to an on-line rough sketch (Fig. 3). Since this sketch of a car is drawn in an on-line environment, tracks of the pen can be obtained. The line fitting process is applied to these tracks instead of the pixels in the image. In an on-line environment, no noise segments are caused by the edge extraction process and thinning process (Minoh et al., 1995).

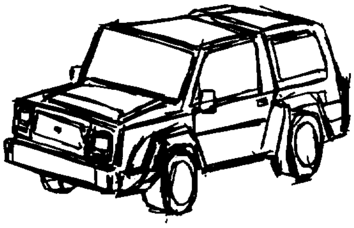
The original image contains 2315 line segments (Fig. 3(a)). 100 straight lines were extracted by SFCE (Fig. 3(b)) and there were 1149 line segments that remained unclustered. The reformed result was made by joining the extracted straight lines and the unclustered line segments (Fig. 3(c)). Therefore, the number of the lines in the reformed result was 1249. The parameters were set as follows: $T_d = 1, T_s = 12°, T_c = 8°, \eta = 100, \sigma = 0.99$. In the reformed result of SFCE, overlapped strokes were clustered into a single line and the sketch was simplified as a whole. However, several overclustered segments were found (e.g. in the wheels of the car), which should be corrected by hand.

The reformed result formed by hierarchical clustering is shown in Fig. 3(d). The number of lines in the reformed result of hierarchical clustering is equal to that of SFCE. We can visually compare the two methods by the frequency of occurence of overclustering, since the difference is obvious: the number of overclustering caused by SFCE is less than that of hierarchical clustering. To reduce overclustering in the hierarchical clustering method, the threshold must be set lower, which will then sacrifice the reforming effect. This result shows that SFCE outperforms hierarchical clustering on reforming a rough sketch.

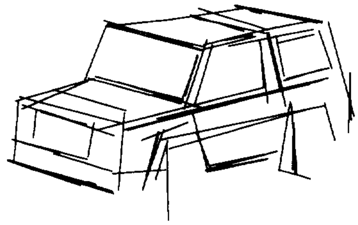
Judging by the ability to extract weakly-collinear segments, SFCE and hierarchical clustering were compared using the synthetic image shown in Fig. 4(a). There is a long line divided into 13 weakly-collinear segments in the center of the image, and 150 widely scattered noise segments. The lengths of the line segments were set to the same value. The direction of the segments belonging to the line differs slightly from its true direction. The difference of the direction of each segment was set randomly from $-7°$ to $7°$. The parameters were set as follows: $T_d = 5, T_s = 20°, T_c = 20°, \eta = 100, \sigma = 0.999$. The task is to extract this long line without extracting noise segments.

When hierarchical clustering was used and the threshold was set at 10 (Fig. 4(b)), the long line was broken into three pieces and four noise inclusions occurred. By increasing the threshold, the noise inclusions can be resolved but the broken lines will never be extracted as one line. For example, when the threshold was set at 14 (Fig. 4(c)), the noise inclusions decreased but the long line was broken into more pieces. Whereas, decreasing the threshold, the long line will be extracted as one line, but the noise inclusions will increase in number. For example, when the threshold was set at 8 (Fig. 4(d)), the long line was extracted as one line, but the noise inclusions increased. In this synthetic image, the noise inclusion could not be avoided by adjusting the threshold without breaking the long line into pieces.

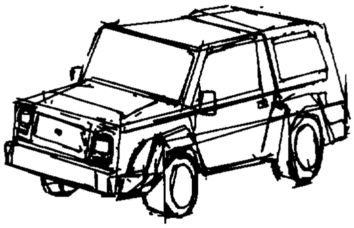
In contrast, using SFCE, the noise inclusion could



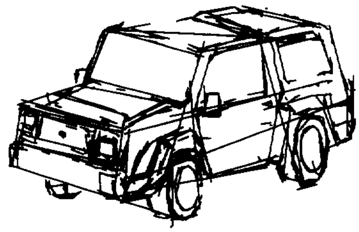
(a) Original Sketch (2315 lines)



(b) Extracted Lines of SFCE

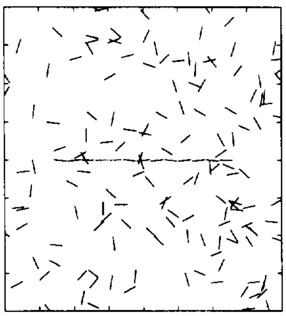


(c) Reformed Result of SFCE (1249 lines)

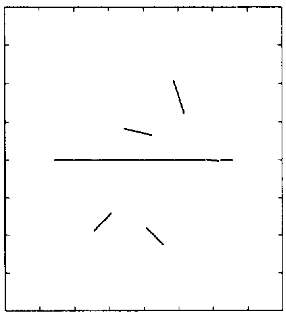


(d) Reformed Result of the Hierarchical Clustering (1249 lines)

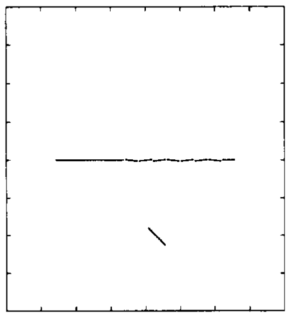
Fig. 3. Line extraction from a rough sketch.



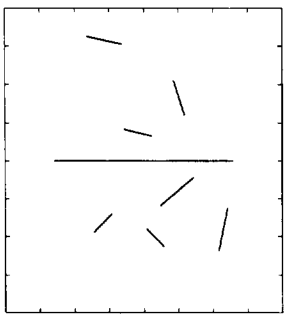
(a) Original Image



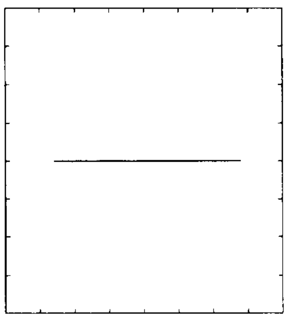
(b) Extracted Lines by Hierarchical Clustering (threshold = 10)



(c) Extracted Lines by Hierarchical Clustering (threshold = 14)



(d) Extracted Lines by Hierarchical Clustering (threshold = 8)



(e) Extracted Lines by SFCE

Fig. 4. Results for a synthetic image with noise.

be avoided (Fig. 4(e)). The long line was extracted as the first cluster, and so no other lines were extracted from noise segments. This result suggests that SFCE is superior to hierarchical clustering in extracting weakly-collinear segments.

6. Conclusion

In this paper, we have proposed a new clustering method called Sequential Fuzzy Cluster Extraction. This method extracts clusters one by one by maximizing a criterion function that is described as the sum of all similarities between members. When it is applied to line extraction, weakly-collinear line segments are clustered as one line, which was very difficult for the conventional threshold-based methods. We have applied the method to rough sketch reforming and confirmed that it outperforms the hierarchical clustering method. SFCE may also be applied to other clustering tasks, especially when much noise is included in the object set to be clustered.

Appendix A

Derivation of Eq. (13) from Eq. (12)

The objective function in Eq. (10) can be rearranged by substituting Eq. (12):

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} e_{ij} x_i x_j - \frac{1}{k-1} \sum_{t=1}^{k-1} \beta_t \sum_{i=1}^{n} w_{ti}^2 x_i^2 \tag{16}$$

$$= Q(\boldsymbol{x}) - \frac{1}{k-1} \sum_{t=1}^{k-1} \beta_t \, dup(\boldsymbol{w}_t, \boldsymbol{x}). \tag{17}$$

So, the optimization problem of Eq. (10) can be rewritten as the following optimization problem: Maximize

$$Q(\boldsymbol{x}) - \frac{1}{k-1} \sum_{t=1}^{k-1} \beta_t \, dup(\boldsymbol{w}_t, \boldsymbol{x}) \tag{18}$$

subject to the constraints

$$\sum_{i=1}^{n} x_i^2 = 1. \tag{19}$$

Comparing it with Eq. (8), this optimization problem does not have the constraint $x_i \geqslant 0$. If the elements of the optimal solution of Eq. (18) are all positive, the optimization problems of Eq. (7) and of Eq. (18) are exactly the same.

The value of the second term in Eq. (16) does not depend on the sign of x_i. Since $e_{ij} \geqslant 0$, all x_i's should have the same sign to maximize the first term. Thus, all the elements of the optimal solution $\boldsymbol{x}^*$ have the same sign. To satisfy the constraint $x_i \geqslant 0$, all the elements must be positive. But, even if all the elements are negative, the optimal solution with the positive elements can be obtained simply by changing signs.

Therefore, the optimal solution of Eq. (7) (i.e. $\boldsymbol{w}_k^*$) can be obtained by making the signs of all the elements of $\boldsymbol{x}^*$ positive. Since $\boldsymbol{x}^*$ can be obtained as the first vector z_1 of H,

$$w_{ki}^* = |z_{1i}|. \tag{20}$$

References

Bellman, R. (1970). *Introduction to Matrix Analysis*, Second Edition. McGraw-Hill, New York.

Burns, J.B., A.R. Hanson and E.M. Riseman (1986). Extracting straight lines. *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (4), 425–455.

Cullum, J. and R.A. Willoughby (1985). A survey of Lanczos procedures for very large real 'symmetric' eigenvalue problems. *J. Comput. Appl. Math.* 12–13, 37–60.

Minoh, M., T. Yamashita and K. Ikeda (1995). Automated reforming of an on-line sketch based on perceptual organization. *6th IFSA World Congress* 1, 661–664.

Mohan, R. and R. Nevetia (1992). Perceptual organization for scene segmentation and description. *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (6), 616–635.

Nacken, P.F.M. (1993). A metric for line segments. *IEEE Trans. Pattern Anal. Mach. Intell.* 15 (12), 1312–1318.