PAPER

RK-Means Clustering: K-Means with Reliability

Chunsheng HUA[†*], **Qian CHEN**[†], ***Nonmembers***, **Haiyuan WU**[†a)],
and* Toshikazu WADA**[†], ***Members

SUMMARY This paper presents an RK-means clustering algorithm which is developed for reliable data grouping by introducing a new reliability evaluation to the K-means clustering algorithm. The conventional K-means clustering algorithm has two shortfalls: 1) the clustering result will become unreliable if the assumed number of the clusters is incorrect; 2) during the update of a cluster center, all the data points belong to that cluster are used equally without considering how distant they are to the cluster center. In this paper, we introduce a new reliability evaluation to K-means clustering algorithm by considering the triangular relationship among each data point and its two nearest cluster centers. We applied the proposed algorithm to track objects in video sequence and confirmed its effectiveness and advantages.

key words: *robust clustering, reliability evaluation, K-means clustering, data classification*

1. Introduction

Clustering algorithms can partition a data set into c groups to reveal its nature structure.

K-means (KM) [25] is a “Hard” algorithm that unequivocally assign each vector in the data set into one of c subsets, where c is a natural number. Given a data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, the K-means algorithm computes c prototypes $\mathbf{w} = (\mathbf{w}_c)$ which minimize the average distance between each vector and its closest prototype:

$$E(\mathbf{w}) = \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{w}_{s_i(\mathbf{w})})^2, \tag{1}$$

where $s_i(\mathbf{w})$ denotes the subscript of the closest prototype to vector $\mathbf{x}_i$. The prototypes can be computed iteratively with the following equation until $\mathbf{w}$ reaches a fixed point.

$$\mathbf{w}_k^{'} = \frac{1}{N_k} \sum_{i:k=s_i(\mathbf{w})} \mathbf{x}_i, \tag{2}$$

K-means clustering has been widely applied to the image segmentation [22]–[24]. Recently, it has also been used for object tracking [17], [18], [20].

Fuzzy and possibilistic clustering generate a membership matrix $\mathbf{U}$, where its elements u_{ik} indicates that the membership of $\mathbf{x}_k$ in cluster i. The Fuzzy C-Means algorithm (FCM) [1], [2], [16] is the most popular fuzzy clustering algorithm. It assumes that the number of clusters c, is known as a priori, and minimizes

$$E_{fcm} = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} \|\mathbf{x}_k - \mathbf{w}_i\|^2, \tag{3}$$

subject to the n probabilistic constrains,

$$\sum_{i=1}^{c} u_{ik} = 1; k = 1, \ldots, n, \tag{4}$$

here, $m > 1$ is the fuzzifier. FCM provides the matrices $\mathbf{U}$ and $\mathbf{W}$ ($\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_c)$) that indicate the membership of each vector in each cluster and the center of each cluster respectively. The conditions for local extreme for Eqs. (3) and (4) are,

$$u_{ik} = \left(\sum_{j=1}^{c} \sqrt[m-1]{\frac{d_{ik}^2}{d_{jk}^2}} \right)^{-1}, \tag{5}$$

and

$$\mathbf{w}_i = \frac{\sum_{k=1}^{n} u_{ik}^{m} \mathbf{x}_k}{\sum_{k=1}^{n} u_{ik}^{m}}. \tag{6}$$

In many applications of K-means or FCM clustering, the data set contains noisy vectors. Using the above clustering methods, every vector in the data set is assigned to one or more clusters, even if it is a noisy one. These clustering methods are not able to distinguish the noisy vectors from the rest vectors in the data set, and those noisy vectors will attract the cluster centers towards them. Therefore, these clustering methods are sensitive to noise. A good K-means (or C-means) clustering method should be robust so that it can determine good clusters for noisy data sets. Several robust clustering methods have been proposed [3]–[15], and a comprehensive review can be found in [3].

The Noise Cluster Approach [4] introduces an additional cluster called noise cluster to collect the outliers (an outlier is a distant vector from all the clusters). The distance between any vectors and the noise cluster is assumed to be the same. This distance can be considered as a threshold. If the distance between a vector and all the clusters is longer than this threshold, it will be attracted to the noise cluster thus classified as an outlier. When all the c clusters have about the same sizes and the size is known, the noise cluster

Manuscript received May 2, 2006.
Manuscript revised December 5, 2006.

[†] The authors are with the Faculty of System Engineering, Wakayama University, Wakayama-shi, 640–8510 Japan.

[*] Presently, with the Institute of Scientific and Industrial Research, Osaka University.

a) E-mail: wuhy@sys.wakayama-u.ac.jp

DOI: 10.1093/ietisy/e91–d.1.96

approach is very effective. When the clusters have different sizes or the size is unknown, this approach will not perform well because it only has one threshold and there is not a method to find a good threshold value for a given data set.

The Possibilistic C-Means Algorithm (PCM) [5], [6] computes a matrix of possibilistic membership $\mathbf{T}$. The element t_{ij} of $\mathbf{T}$ indicates the possibilistic membership of vector $\mathbf{x}_j$ belongs to i^{th} cluster. Although PCM seems better than FCM and hard K-means clustering approach, it often finds identical clusters.

Jolion [13] proposed an approach called the Generalized Minimum Volume Ellipsoid Method (GMVE). In the GMVE approach, one finds a minimum volume ellipsoid that covers (at least) h vectors of the data set $\mathbf{X}$. After that, it finds the best cluster by reducing the value of h gradually. This approach is computationally expensive and needs to specify several threshold parameters. Moreover, when the cluster shapes to be detected cannot be described by ellipsoids, this approach would not work.

Chintalapudi [10] proposed an approach called Credibility Fuzzy C-Means method (CFCM). It assigns a credibility value for each vector $\mathbf{x}_k$ according the ratio of the distance between it and its closest cluster to the distance between the farthest vector and its closest cluster. If the farthest outlier is much farther than the rest outliers, this approach will assign high credibility values to most of the outliers. In this case, CFCM would not work well.

2. RK-Means Clustering

2.1 Reliability

Both the K-means and fuzzy C-means clustering (FCM) algorithm (including its extensions) assume that the number (c) of the clusters of a data set is known. In the real world, a data set may contain many noisy vectors and the number of clusters of it is often unknown. The noisy vectors and the data vectors that do not belong to any of the assumed clusters are often very distant from any of the c prototypes (thus called outliers), so it would not be meaningful to assign them a cluster number for K-means algorithm or a high membership value to any of the c clusters for FCM.

We attempt to decrease the outlier sensitivity in K-means clustering by introducing a new variable *reliability* to distinguish an outlier from a non-outlier. Since outliers are distant from any of the c prototypes, the distance between a data vector and its closest prototype should be taken into account in order to tell whether the vector is an outlier or not. Existing methods such as noise cluster approach [4] or credibility fuzzy C-means algorithm use that distance to tell an outlier from a non-outlier. However, the distance between a data vector and its nearest cluster center can not be a measure of outlier by itself. To tell if a data vector is an outlier or not, one should consider both the distance between a data vector and its nearest cluster center and the structure of the cluster centers, that is the distance between cluster centers.

As shown in Fig. 1, where $\mathbf{w}_1$ and $\mathbf{w}_2$ are two cluster centers, and $\mathbf{x}_1$ and $\mathbf{x}_2$ are two data vectors. In the figure, if we only look at the distance from a vector ($\mathbf{x}_1$ or $\mathbf{x}_2$) to its closest cluster center ($\mathbf{w}_1$), we can only say that $\mathbf{x}_2$ is closer to $\mathbf{w}_1$ than $\mathbf{x}_1$. However, no conclusions can be drawn from the value d_{11} and d_{21} as which of the vectors is more like an outlier. By considering the shape of the triangle $\triangle\mathbf{x}_1\mathbf{w}_1\mathbf{w}_2$ and $\triangle\mathbf{x}_2\mathbf{w}_1\mathbf{w}_2$, that is the relation among three distances $(d_{11}, d_{12}, d_{w12})$, and the one among $(d_{21}, d_{22}, d_{w12})$, we can say that $\mathbf{x}_1$ should be considered as an outlier while $\mathbf{x}_2$ should not be.

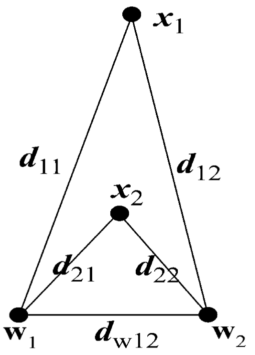


Fig. 1 Introduce the concept of reliability.

In this paper we denote the data set as $\mathbf{X}$, its n vectors as $\{\mathbf{x}_k\}_{k=1}^{n}$ and the cluster centers $\mathbf{w}_i, i = 1, \ldots, c$. We define reliability for a data vector $\mathbf{x}_k$ as

$$\mathcal{R}_k = \frac{||\mathbf{w}_{f(\mathbf{x}_k)} - \mathbf{w}_{s(\mathbf{x}_k)}||}{d_{fk} + d_{sk}}, \tag{7}$$

where

$$\begin{aligned} d_{kf} &= || \mathbf{x}_k - \mathbf{w}_{f(\mathbf{x}_k)} ||, \\ d_{ks} &= || \mathbf{x}_k - \mathbf{w}_{s(\mathbf{x}_k)} ||, \end{aligned} \tag{8}$$

$f(\mathbf{x}_k)$ and $s(\mathbf{x}_k)$ are the subscript of the closest and the secondly closest cluster centers to vector $\mathbf{x}_k$:

$$\begin{aligned} f(\mathbf{x}_k) &= \underset{i=1,\ldots,c}{argmin}(||\mathbf{x}_k - \mathbf{w}_i||), \\ s(\mathbf{x}_k) &= \underset{i=1,\ldots,c,i\neq f}{argmin}(||\mathbf{x}_k - \mathbf{w}_i||). \end{aligned} \tag{9}$$

The farther $\mathbf{x}_k$ is from its nearest cluster center, the lower is its reliability. The value of this reliability is not determined by the distance between $\mathbf{x}_k$ and its nearest cluster center itself. Instead, it is determined by using the distance between the two nearest cluster centers to measure how distant that $\mathbf{x}_k$ from the two nearest cluster centers. This makes the reliability evaluation becoming more reasonable than only use the distance from a data vector to its nearest cluster center.

2.2 Degree of Reversion

For each data vector, fuzzy C-means algorithm computes c memberships that indicate the degrees of the vector belong to the c clusters. This is computationally expensive, especially for real-time video image processing. In this research, we use a simple method to evaluate the degree of a vector belongs to its closest cluster to achieve realistic clustering than "hard" techniques while keeping the computation cost to be low. The degree μ_k of a vector $\mathbf{x}_k$ belongs to its closest cluster is computed from the distance from it to its closest (d_{kf}) and the secondly closest (d_{ks}) cluster centers:

$$\mu_k = \frac{d_{ks}}{d_{kf} + d_{ks}}, \tag{10}$$

Since $\mathcal{R}_k$ – the reliability of $\mathbf{x}_k$ –indicates how reliable that $\mathbf{x}_k$ can be classified, the possibility that $\mathbf{x}_k$ belongs to its closest cluster can be computed as the product of $\mathcal{R}_k$ and μ_k.

$$t_k = \mathcal{R}_k * \mu_k. \tag{11}$$

2.3 RK-Means Clustering Algorithm

RK-means clustering algorithm partitions $\mathbf{X}$ by minimizing the following objective function:

$$J_{rkm}(\mathbf{w}) = \sum_{k=1}^{n} t_k \|\mathbf{x}_k - \mathbf{w}_{f(\mathbf{x}_k)}\|^2, \tag{12}$$

The cluster centers $\mathbf{w}$ can be obtained by solving the equation

$$\frac{\partial J_{rkm}(\mathbf{w})}{\partial \mathbf{w}} = 0 \tag{13}$$

The existence of the solution to Eq. (13) can be proved easily if the Euclidean distance is assumed. To solve this equation, we first compute an approximate $\mathbf{w}$ with the following equation:

$$\mathbf{w}_j = \frac{\sum_{k=1}^{n} \delta_j(\mathbf{x}_k) t_k \mathbf{x}_k}{\sum_{k=1}^{n} \delta_j(\mathbf{x}_k) t_k(\mathbf{x}_k)}. \tag{14}$$

where

$$\delta_j(\mathbf{x}_k) = \begin{cases} 1 & \text{if } j = f(\mathbf{x}_k) \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

Then $\mathbf{w}$ can be obtained by applying Newton's algorithm using the result of Eq. (14) as the initial values.

The RK-means clustering algorithm is summarized as follows:

```
1) Initialization
i) given the number of clusters $c$ and
ii) given an initial value to each cluster center $\mathbf{w}_i, i = 1, \ldots, c$.
The initialization can be done by applying K-means or fuzzy c-means approaches, or performed manually.
2) Iteration
while $\mathbf{w}_i, i = 1, \ldots, c$ do not reach fixed points,
Do
i) calculate $f(\mathbf{x}_k)$ and $s(\mathbf{x}_k)$ for each $\mathbf{x}_k$.
ii) update $\mathbf{w}_i, i = 1, \ldots, c$ by solving Eq. (13).
```

3. Object Tracking Using RK-Means Algorithm

The RK-means can be used for realizing robust object tracking in video sequences. (see Fig. 2). Because the most important thing while object tracking is to classify the unknown pixels into target or background clusters, object tracking can be considered as a binary classification. Therefore, the first and second closest clusters in RK-means clustering algorithm will be replaced by the target and background clusters. Since it is more important to tell if the unknown pixel belongs to the target cluster or not, the first closest cluster will be considered as the target cluster, and naturally second closest cluster is the background cluster.

Each pixel within the search area will be classified into target and background clusters with the RK-means clustering algorithm. Noise pixels that neither belong to target nor background clusters will be given low reliability and ignored. Pixels belong to the target group is farther divided into N target clusters, each of them describes the target pixels having similar color. The pixels belonging to the background clusters is also divided into m background clusters.

We use a 5D uniform feature space to describe image features. In the 5D feature space, both the color and the position of a pixel is described by a vector $\mathbf{f} = [\mathbf{c}\ \mathbf{p}]^T$ uniformly. Here $\mathbf{c} = [Y\ U\ V]^T$ describes the color and $\mathbf{p} = [x\ y]^T$ describes the position on image plane.

In the 5D feature space, we describe the target centers as

$$\mathbf{f_T}(i) = [\mathbf{c_T}(i)\ \mathbf{p_T}(i)]^T, (i = 1 \sim N),$$

N is the number of target clusters. Each target cluster describes the target pixels having similar color. The background pixels are represented as

$$\mathbf{f_B}(j) = [\mathbf{c_B}(j)\ \mathbf{p_B}(j)]^T, (j = 1 \sim m),$$

m is the number of the background clusters. An *unknown* pixel is described by $\mathbf{f}_u = [\mathbf{c}_u\ \mathbf{p}_u]^T$. Thus we can get the subscript of the closest target and background cluster centers to $\mathbf{f}_u$ as follows.

$$\begin{aligned} s_T(\mathbf{f}_u) &= \underset{i=1\sim N}{argmin}\{\|\mathbf{f_T}(i) - \mathbf{f}_u\|\}, \\ s_B(\mathbf{f}_u) &= \underset{j=1\sim m}{argmin}\{\|\mathbf{f_B}(j) - \mathbf{f}_u\|\}. \end{aligned} \tag{16}$$

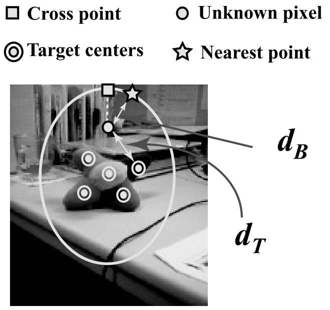
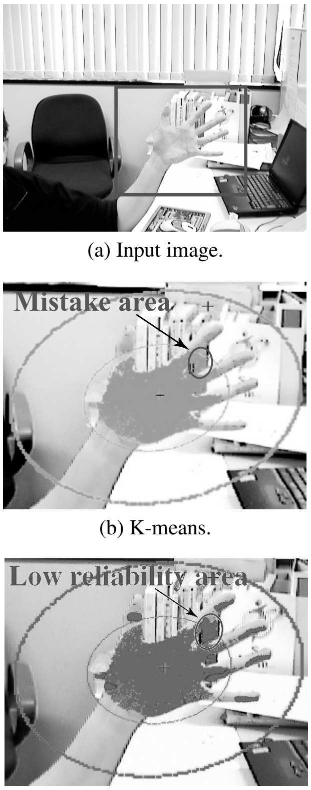


Fig. 2 Explanation for target clustering with multiple colors.



(a) Input image.

(b) K-means.

(c) RK-Means: The red area has low reliability.

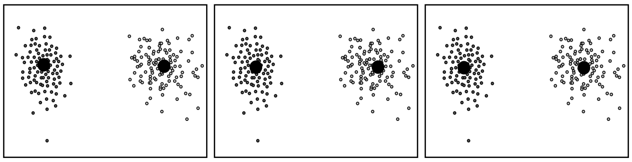
Fig. 3 A comparative result of object tracking with the K-Means and RK-Means algorithms.

Since $\mathbf{f}_u$ is requested to be classified into target or background clusters, it can be considered that only two candidate clusters (target and background clusters) exist while object tracking. Therefore, obviously one cluster will be the first nearest cluster, and another will be the second nearest one. Because it is more important to check if $\mathbf{f}_u$ is a target pixel or not, $s_{\mathbf{T}}$ is considered as the first nearest cluster and $s_{\mathbf{B}}$ as the second nearest cluster. By applying $s_{\mathbf{T}}$ and $s_{\mathbf{B}}$ to Eqs. (9), (8), the RK-means algorithm can be used to remove the noise data (as shown in Fig. 3) and update the target centers $f_{\mathbf{T}}(i), i = 1, \ldots, N$. Because the background clusters are defined and selected from the ellipse contour and updating background clusters with Eq. (14) will move them from the ellipse contour, we update the background clusters (also the ellipse contour) by the method mentioned in [17].

4. Experiment and Discussion

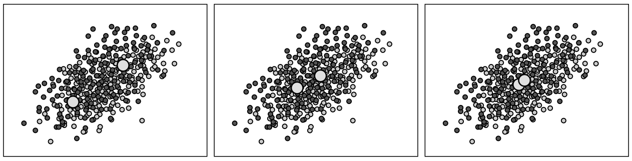
4.1 Evaluating the Efficiency of RKM

In the following parts of this paper, we abbreviate the Hard (or conventional) K-means clustering as CKM, Fuzzy C-means as FKM and the reliability K-means as RKM. To evaluate the proposed RKM algorithm, we compare it with the CKM and FKM clustering under different conditions, in the following illustrations the large "•" represents the target center. In Figs. 4, 5, the initial value of K is two (one cluster is indicated by red, another by green), the position of initial points is the same and all the algorithms run at the same iterations.



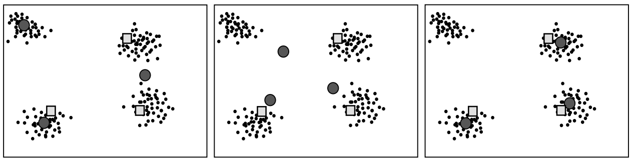
(a) CKM. (b) FKM. (c) RKM.

Fig. 4 Clustering with separated clusters.



(a) CKM. (b) FKM. (c) RKM.

Fig. 5 Clustering with completely overlapped clusters.



(a) CKM. (b) FKM. (c) RKM.

Fig. 6 Clustering with missing cluster problem.

In Fig. 4, since the distribution of two clusters is well separated, all methods give the similar good result.

In Fig. 5, the distributions of two clusters merge as one cluster. In such case, it is reasonable to consider them as one cluster. The CKM and FKM algorithm still brutally divide it into two separated clusters. With Eq. (7), the proposed RKM can gradually move the initial centers together and give the result as shown in (c). Although the result of our RKM algorithm works better than the CKM and FKM methods, the whole clustering procedure becomes unreliable. That is because the cluster centers tend to move together, thus Eqs. (11), (7) will produce low reliability for each data item. In our future work, we consider the function of merging is necessary for our RKM algorithm.

Figure 6 shows a case where four clusters exist but the initial value of K is three (hereafter we call the phenomenon like this as the missing cluster problem). Here, the yellow "□" denotes the initial position of cluster centers which is manually defined, and the red "•" means the final clustering result. The CKM only puts one cluster center correctly and makes the completely wrong results of the other two cluster centers. The FKM is heavily affected by the missing cluster, because it brutally allocates a membership to each input data item. With Eqs. (7), (11), our RKM correctly classifies the initial three centers by giving an extremely low reliability to the missing cluster.

In Fig. 7, we show the comparative experimental result

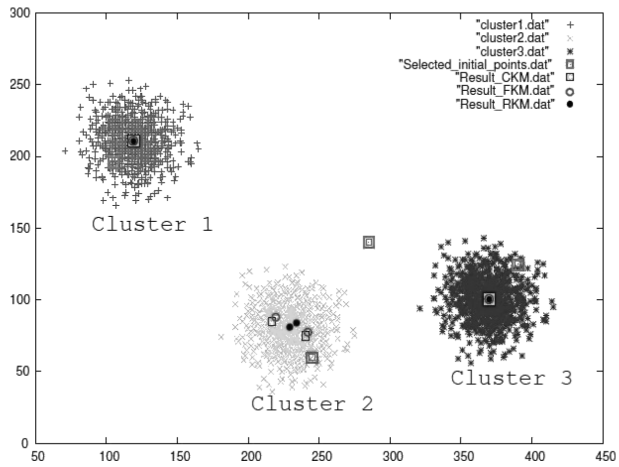


Fig. 7 Experiment when the given number of K is larger than the real number of clusters. Cluster 1: red points; Cluster 2: green points; Cluster 3: blue points. All three clustering algorithms give the same results to Cluster 1 and 3. They give different conclusion on Cluster 2.

Table 1 When the given number of K is larger than the real number of clusters, the clustering result of CKM, FKM and RKM for cluster 2. The real center of cluster 2 is located at (230, 80).

	Point 1	Point2
Initial Given Position	(285, 140)	(245, 60)
Result of CKM	(240, 75)	(218, 86)
Result of FKM	(238, 75)	(220, 86)
Result of RKM	(235, 84)	(229, 81)

of the CKM, FKM and RKM in the case that the given initial number of K ($K = 4$) is larger than that of the real input data (3 clusters). In this image, the pink hollow "□" is the initial given point and the solid "□" of sky blue color, the red hollow "○" and the black "•" denote the final result of CKM, FKM and RKM clustering algorithms, respectively. In this experiment, three of the initial points are correctly located within the corresponding clusters, but one additional initial point is assigned to the right-top of cluster 2. In such case, because both the CKM and FKM will allocate one data to one or several clusters without considering if the clustering process for such data is reliable or not, either of them will assign some data of cluster 2 to that additional initial point. Thus, their clustering result is heavily affected. As for the RKM algorithm, it becomes insensitive to the initial given points by evaluating the reliability of clustering data to the additional initial point. And finally, the additional cluster center is removed towards the real center of cluster 2. Although the result of RKM when clustering data to cluster 2 is not the perfect result, compared with CKM and FKM, the RKM gives the best result.

Through Table 1, we can see that there is just a little difference between the result of CKM and that of FKM. Both of them are heavily affected by the additional given initial point (285, 140) which is located out of cluster 2. Although CKM and FKM try to merge the two given centers into one cluster, the performance of them is not satisfying. Contrast to CKM and FKM, although the proposed RKM does not really convert the two initial points into one cluster, it seems to merge them most successfully.

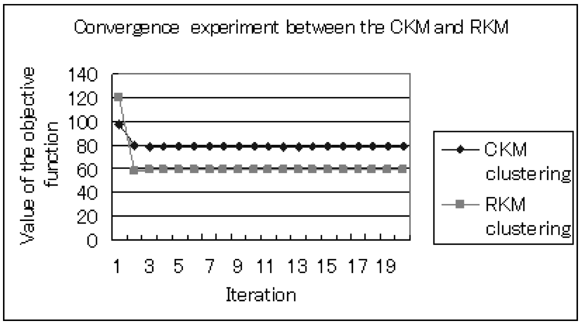


Fig. 8 Convergence of the CKM and RKM algorithms.

4.2 Convergence of the RKM

One important characteristic of the RKM clustering algorithm is whether it converges or not. As for this question, the objective function of RKM is a good index to check whether it converges or not. If the value of the objective function remains unchanged (or the changes are so small that such they can be ignored) after a finite iteration number, that denotes that the RKM algorithm converges. Otherwise, the RKM diverges.

To perform this experiment of convergence, we apply the RKM to the IRIS dataset to check its convergence. The IRIS dataset has 150 data points. It is divided into three groups and two of them are overlapping. Each group contains 50 data points. Each point has four attributes.

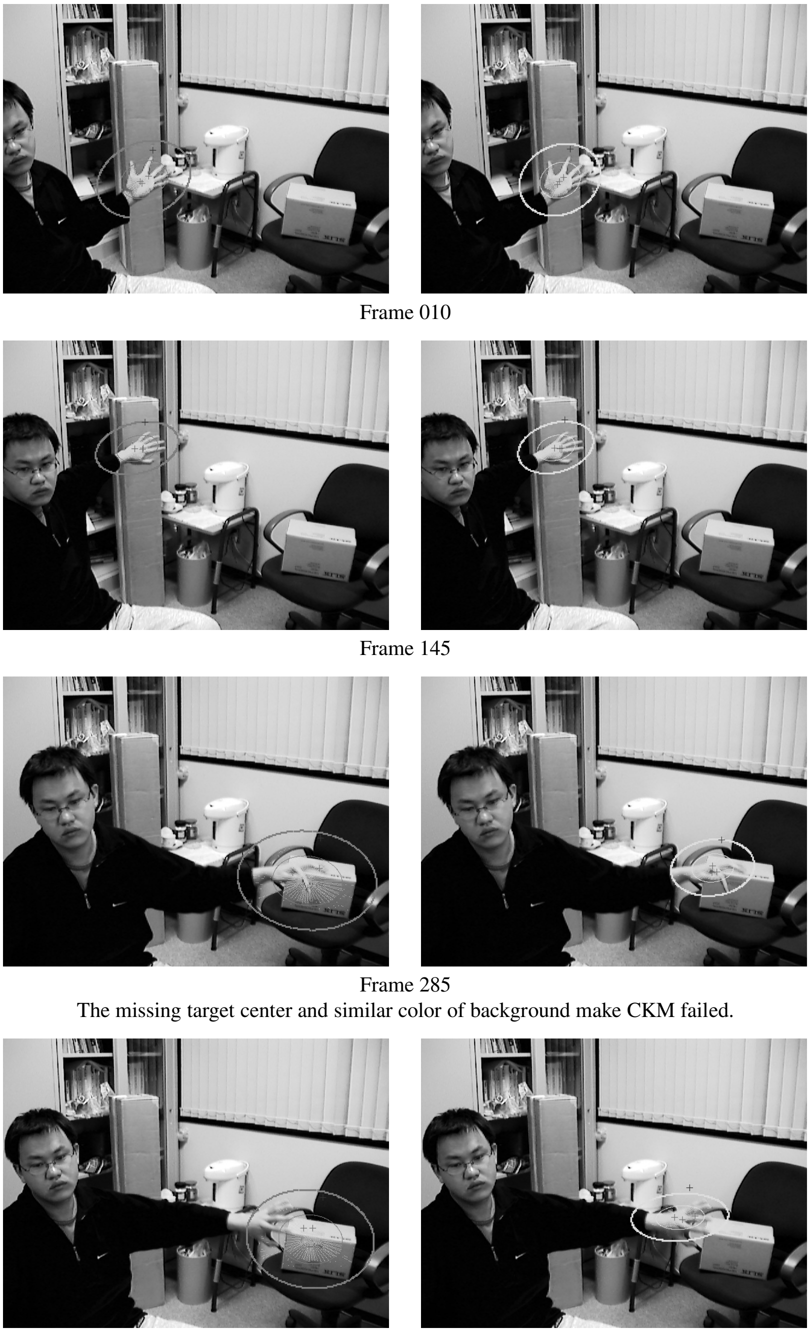
In Fig. 8, because the objective functions of the CKM and RKM are different from each other, we can not say that the RKM converges better than the CKM, but we could say that the RKM algorithm could converge as well as the CKM algorithm. Since the RKM adds the reliability estimation to the clustering process, as the speed of convergence, it can converge faster than the CKM algorithm.

4.3 Object Tracking Experiment

4.3.1 Manual Initialization

It is necessary to assign the number of clusters when using the K-means algorithm to classify a data set. In our tracking algorithm, in the first frame, we manually select N points on the object to be tracked and use them as the N initial target cluster centers. We let the initial ellipse (search area) be a circle. The center of circle is put at the centroid of the N initial target cluster centers. We manually select one point out of the object and let the circle cross it. In the following frames, the ellipse center is updated according to the result of target detection.

Here, we use m representative background samples selected from the ellipse contour. Theoretically, it is best to use all pixels on the boundary of the search area (ellipse contour) as representative background samples. However,



Frame 010

Frame 145

Frame 285

The missing target center and similar color of background make CKM failed.

In Frame 305

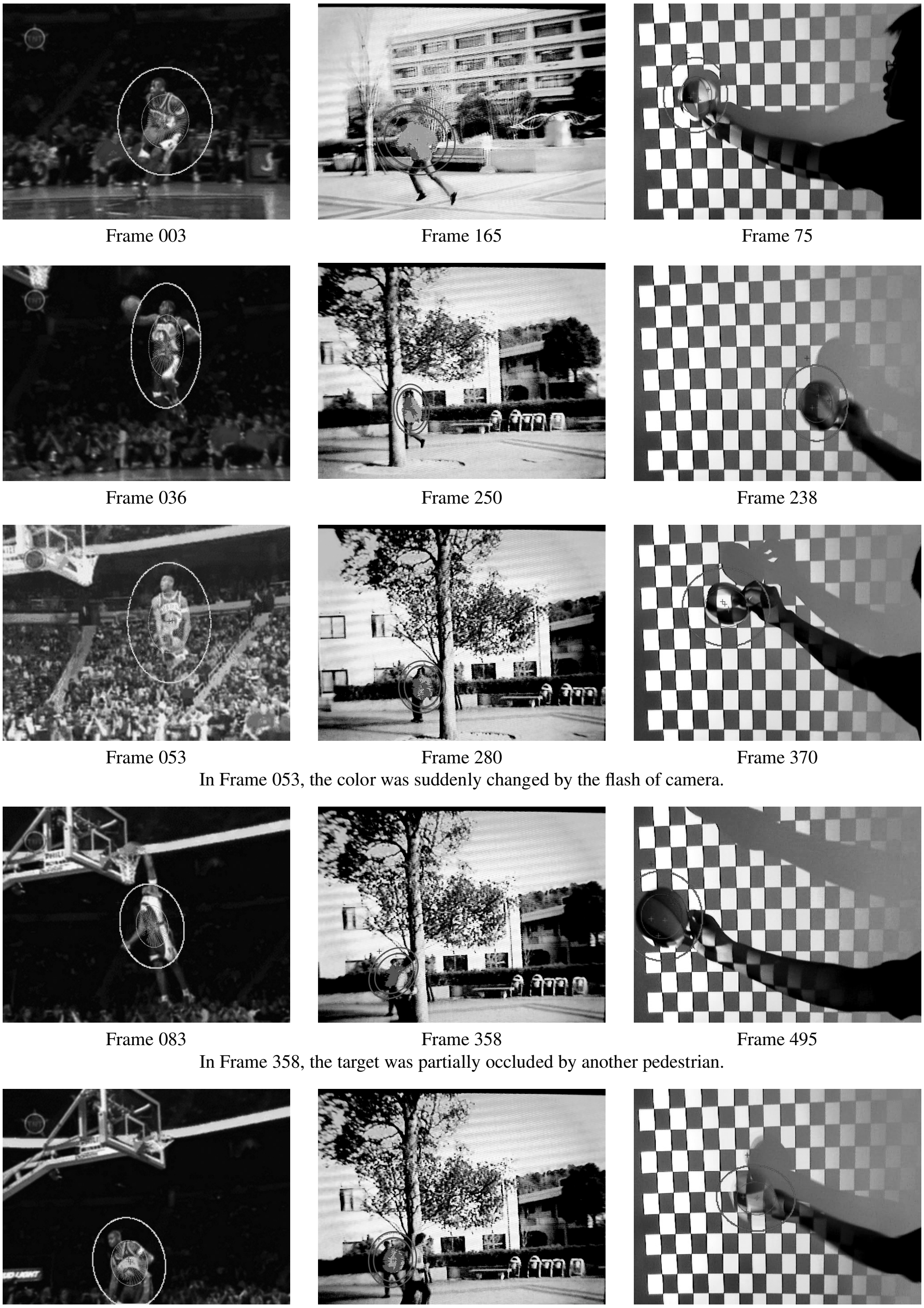
(a) CKM-based tracking.

(b) RKM-based tracking.

Fig. 9 Comparative experiment with CKM and RKM under complex background.

since there are many pixels on the ellipse contour (several hundreds in common cases), the number of the clusters will become a huge one. This dramatically reduces the processing speed of pixel classification during tracking. In order to realize real-time processing speed, we let m be a small number. Through extensive experiment of tracking objects of wide classes, we found that $m = 9$ is a good choice for fast tracking while keeping the stable tracking performance. Of the 9 points, 8 points are resolved by the 8-equal division of the ellipse contour, and the 9^{th} one is the cross point between the ellipse contour and the line connecting the pixel to be classified and the center of the ellipse center.

The target in Fig. 9 is a hand. During tracking, the hand freely changes between the palm and the back. Although the colors of palm and back of a hand look similar, they are not exactly the same colors. At the beginning of this comparative experiment, the hand is showing the palm, so the initial target colors are those of the palm. Such object is difficult



Frame 003 Frame 165 Frame 75

Frame 036 Frame 250 Frame 238

Frame 053 Frame 280 Frame 370

In Frame 053, the color was suddenly changed by the flash of camera.

Frame 083 Frame 358 Frame 495

In Frame 358, the target was partially occluded by another pedestrian.

Frame 110 Frame 361 Frame 673

In Frame 673, the target was blurred because of its high-speed movement.

Fig. 10 Experiment result with the RKM tracking algorithm. The left column shows a case with the non-rigid object. The middle column shows a result with scaling and occlusion. The right one shows the experiment with illumination variance.

for tracking because: 1) many background areas have quite similar color to the target; 2) the arbitrary deformation of non-rigid target shows different shape in the image. To evaluate the performance of the CKM and RKM fairly, the two tracking algorithm are tested with the same sequence, given the same initial points, and both CKM and RKM are run at the same iterations.

As for the CKM-based tracking algorithm [17] in the left column, the tracking failed since frame 285. In frame 285, the hand turned to show the back and the interfused background contained quite similar color to the initial target color (the color of the palm selected from the first frame). As mentioned in Fig. 6 (a), the CKM-based tracking algorithm greatly suffers from the missing cluster problem. Since the background interfusion and missing cluster happened simultaneously, the CKM-based tracking algorithm mistakenly took the interfused background area that contains similar color to the target as the target area. So the search area was wrongly updated, which caused the CKM-based tracking algorithm to fail in frame 305. As for the RKM-based tracking algorithm in the right column, as mentioned in Fig. 6 (c), the RKM-based tracking algorithm is insensitive to the missing cluster problem by assigning the missing cluster with a low reliability. Therefore, in frame 285, the RKM-based tracking algorithm could correctly detect the target area and update the search area, which ensures the robust object tracking.

We also applied the proposed RKM-based tracking algorithm to many other sequences. Figure 10 shows some experiment results, which indicates our tracking algorithm can deal with the deformation of non-rigid object, color variance caused by the illumination changes and occlusion. Moreover, this tracking algorithm can also deal with the rotation and size change of the target.

All the experiments were performed with a desktop PC with 3.06 GHz Intel XEON CPU, and the image size was 640×480 pixels. When the target size varied from $140 \times 140 \sim 200 \times 200$ pixels, the processing speed of our algorithm was about $9 \sim 15$ ms/frame.

5. Conclusion

In this paper, we presented a reliability-based K-means clustering algorithm which is insensitive to the initial value of K. By applying the proposed algorithm to object tracking, we realized a robust tracking algorithm. Through the comparative experiments with the CKM-based tracking algorithm [17], we confirmed the proposed RKM-based tracking algorithm can work more robustly when the background contains similar colors to the target.

Besides object tracking, we also confirmed the proposed RKM clustering algorithm can be applied to image segmentation.

Acknowledgement

This research is partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Scientific Research (A)(2), 16200014 and (C)18500131 and (C)19500150.

References

[1] J.C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," Journal of Cybernetics, vol.3, pp.32–57, 1973.

[2] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithm, Plenum Press, New York, 1981.

[3] R.N. Dave and R. Krishnapuram, "Robust clustering methods: A unified review," IEEE Trans. Fuzzy Syst., vol.5, no.2, pp.270–293, May 1997.

[4] R.N. Dave, "Characterization and detection of noise," Pattern Recognit. Lett., vol.12, no.11, pp.657–664, 1991.

[5] R. Krishnapuram and J.M. Keller, "A possibilistic approach to clustering," IEEE Trans. Fuzzy Syst., vol.1, no.2, pp.98–110, 1993.

[6] R. Krishnapuram and J.M. Keller, "The possibilistic c-means algorithm: Insights and recommendations," IEEE Trans. Fuzzy Syst., vol.4, no.3, pp.98–110, 1996.

[7] A. Schneider, "Weighted possibilistic c-means clustering algorithms," Ninth IEEE International Conference on Fuzzy Systems, vol.1, pp.176–180, May 2000.

[8] N.R. Pal, K. Pal, J.M. Keller, and J.C. Bezdek, "A new hybrid c-means clustering model," IEEE International Conference on Fuzzy Systems, vol.1, pp.179–184, July 2004.

[9] N.R. Pal, K. Pal, J.M. Keller, and J.C. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," IEEE Trans. Fuzzy Syst., vol.13, no.4, pp.517–530, Aug. 2005.

[10] K.K. Chintalapudi and M. Kam, "A noise-resistant fuzzy C means algorithm for clustering," 1998 IEEE International Conference on Fuzzy Systems, vol.2, pp.1458–1463, May 1998.

[11] J.-L. Chen and J.-H. Wang, "A new robust clustering algorithm-density-weighted fuzzy c-means," IEEE International Conference on Systems, Man, and Cybernetics, vol.3, pp.90–94, 1999.

[12] J.M. Leski, "Generalized weighted conditional fuzzy clustering," IEEE Trans. Fuzzy Syst., vol.11, no.6, pp.709–715, 2003.

[13] J.M. Jolion, P. Meer, and S. Bataouche, "Robust clustering with applications in computer vision," IEEE Trans. Pattern Anal. Mach. Intell., vol.13, no.8, pp.791–802, Aug. 1991.

[14] M.A. Egan, "Locating clusters in noisy data: A genetic fuzzy c-means clustering algorithm," 1998 Conference of the North American Fuzzy Information Processing Society, pp.178–182, Aug. 1998.

[15] J.-S. Zhan and Y.-W. Leung, "Robust clustering by pruning outliers," IEEE Trans. Syst. Man Cybern. B, Cybern, vol.33, no.6, pp.983–998, 2003.

[16] J.J. DeGruijter and A.B. McBratney, "A modified fuzzy K-means for predictive classification," in Classification and Related Methods of Data Analysis, ed. H.H. Bock, pp.97–104, Elsevier Science, Amsterdam, 1988.

[17] C. Hua, H. Wu, T. Wada, and Q. Chen, "K-means tracking with variable ellipse model," IPSJ Tans. CVIM, vol.46, no.Sig 15(CVIM12), pp.59–68, 2005.

[18] T. Wada, T. Hamatsuka, and T. Kato, "K-means tracking: A robust target tracking against background involution," MIRU2004, vol.2, pp.7–12, 2004.

[19] H.T. Nguyen and A. Semeulders, "Tracking aspects of the foreground against the background," ECCV, vol.2, pp.446–456, 2004.

[20] B. Heisele, U. Kreßel, and W. Ritter, "Tracking non-rigid moving objects based on color cluster flow," CVPR, pp.253–257 1997.

[21] J. Hartigan and M. Wong, "Algorithm AS136: A k-means clustering algorithm," Applied Statistics, vol.28, pp.100–108, 1979.

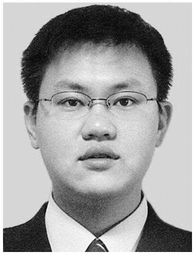
[22] F. Gibou and R. Fedkiw, "A fast hybrid k-means level set algorithm for segmentation," 4th Annual Hawaii International Conference on Statistics and Mathematics, pp.281–291, 2005.

[23] C. Baba, et al., "Multiresolution adaptive K-means algorithm for segmentation of brain MRI," ICSC 1995, pp.347–354, 1995.

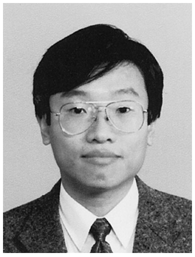
[24] P.K. Singh, "Unsupervised segmentation of medical images using DCT coefficients," VIP 2003, pp.75–81, 2003.

[25] J.B. MacQueen, "Some methods for classification and analysis of multivariate observations," Proc. 5-th Berkeley Symposium on Mathematical Statistics and Probability, vol.1, pp.281–297, Berkeley, University of California Press, 1967.

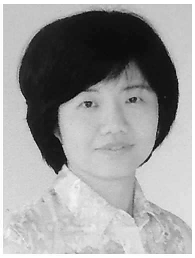
[26] R. Zass and A. Shashua, “A unifying approach to hard and probabilistic clustering,” ICCV, vol.1, pp.294–301, 2005.
[27] A.Y. Ng, M.I. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” NIPS, pp.849–856, 2002.
[28] M. Pavan and M. Pelillo, “A new graph-theoretic approach to clustering and segmentation,” CVPR, pp.145–152, 2003
[29] S.X. Yu and J. Shi, “Multiclass spectral clustering,” ICCV, pp.313–319, 2003.



Chunsheng Hua received his BE degree in electronic engineering from Shenyang University of Technology in 2001. He received his MS degree from the Department of Mechanical and System Engineering at Kyoto Institute of Technology in 2004. He received his Ph.D. degree in computer vision from the graduate school of system engineering at Wakayama University in 2007. He is a student member of IEEE and IPSJ. Now, he is working at the Institute of Scientific Industrial Research of Osaka University as a post doctor. He received the Funai award of IPSJ Digital Courier for young researchers in 2006. His research interests include face detection and recognition, color-based object tracking.



Qian Chen received the Ph.D. degree from Osaka University in 1992. From 1992–1994, he was a researcher at the Laboratory of Image Information and Science and Technology. From 1994–1995, he was the research associate at Osaka University. From 1995–1997, he was the research associate at the Nara Institute of Science and Technology. Since 1997, he has been the associate professor at Wakayama University. He is a member of IPSJ and RSJ.



Haiyuan Wu received the Ph.D. degree from Osaka University in 1997. From 1996–2002, she was the research associate at Kyoto Institute of Technology. Since 2002, she has been an associate professor at Wakayama University. She is a member of IPSJ, ISCIE and the Human Interface.

Toshikazu Wada received his BEng degree in electronic engineering from Okayama University, MEng degree in computer science from Tokyo Institute of Technology and DEng degree in applied electronics from Tokyo Institute of Technology, in 1984, 1987, 1990, respectively. He is currently a professor in Department of Computer and Communication Science, Wakayama University. His research interests include pattern recognition, computer vision, image understanding and artificial intelligence. He received the Marr Prize at the International Conference on Computer Vision in 1995, Yamashita Memorial Research Award from the Information Processing Society Japan (IPSJ), and the Excellent Paper Award from the Institute of Electronics, Information, and Communication Engineerings (IEICE), Japan. He is a member of the IPSJ, the Japanese Society for Artificial Intelligence and IEEE.