UDK 004.932.2

S.I. Bogucharskiy, A.G. Kagramanyan, O.D. Mikhnova

IMAGE SEGMENTATION WITH FUZZY J-MEANS METHOD

Abstract. *The reason for novel method development consists in absence of existing ones that could quickly cope with undefined overlapping image segments without trapping into local extremum. As image segmentation is closely related to clustering, boundaries between image segments are decided to be presented using fuzzy clusters. Crisp and fuzzy solutions to image processing are observed in this paper along with matrix modification of fuzzy J-means clustering that is proposed based on modified fuzzy C-means algorithm.*

Keywords: *image segmentation, crisp clustering, fuzzy clustering, C-means, J-means matrix modification*

Introduction

Spatial segmentation is an important phase in many image and video processing applications. It assumes partitioning the whole field of view into uniform, in a sense, areas of interest. The union of such neighbouring partitions appears to be different from the specified point of view. Therewith, vector space may be presented using visual or geometric features obtained from image pixel information analysis. Among a large variety of image segmentation techniques, clustering is considered one of the most effective [1].

Clustering multidimensional observations is one of the key problems from a wide research area known as ‘data mining’. The challenges faced here are as follows: different distribution law of initial data that is a priori unknown, necessity in real time or near real time processing, variety of cluster shapes and their overlapping, existence of noise and outliers, without mentioning difficulties in initial partitioning and similarity metric choice. Many methods have been already developed for solving these problems, starting from strictly mathematically formalized and up to heuristic based ones [2, 3]. Despite huge efforts in this field, the aforementioned problems are left partially unsolved.

The most frequently used approach to clustering problem consists in partitioning feature vector space implemented by finding p vectors of centroids $C(l)$, $l = 1, 2, \ldots, p$ around which clustered sample vectors are grouped. Well-known methods, there, with extended mathematical grounding are those connected with sum of squares minimization (in terms of Euclidean metric) of intra-cluster distances between vectors of observations and calculated centroids. Among centroid based methods, H-means, K-means, C-means, J-means and their modifications [2-7] have

© BogucharskiyS.I., KagramanyanA.G., MikhnovaO.D., 2014

gained great popularity.

Despite of their effectiveness, centroid based methods (such as K-means, H-means and fuzzy C-means) possess the same drawback: they tend to stuck at local extremum during optimization process. This problem occurs quite often for multiextremal optimization, and to overcome it, there are several approaches: from simple search restart or global random search application [8] and up to using more complicated genetic or immune algorithms. Unfortunately, these approaches significantly increases time needed for multimedia information processing that is unallowable while working with video data bases containing huge amounts of visual information.

Vector clustering procedures with global properties are J-means method for crisp clustering [5] and its fuzzy case FJM (abbreviated from 'fuzzy J-means') [9]. When optimization procedure traps into a local extremum, these methods accomplish 'jumps' in the vicinity of the extremum, which leads the procedure out of the trap to the attraction of a more 'deep' extremum. In further sections, crisp and fuzzy image segmentation via clustering is observed along with matrix modification of fuzzy J-means method that is proposed and discussed in application to image segmentation.

1. Crisp and Fuzzy Image Segmentation

Boundaries between image segments may be of two types. They can be presented as strict separate objects and overlapping objects. The latter is even more practicable because real objects usually overlap each other and the background, i.e. they are not clearly defined. The same thing is true for crisp and fuzzy clustering. In the first case, cluster boundaries are strict and one observation (image pixel, for instance) is related to one and only one cluster, i.e. clusters are mutually exclusive. In fuzzy clustering, some observations may belong to more than one cluster with membership value specified [1].

Consider crisp form of image segmentation, during which an image given by $(M \times N)$-matrix containing pixel information is partitioned into uniform, in a sense, classes (segments or clusters). With this proviso, the initial image matrix is divided into 'window' blocks of size $(m \le M) \times (n \le N)$. Then, after each of these blocks is converted (vectorized) into $(mn \times 1)$-vectors of amount $N^* = MN(mn)^{-1}$, any of the aforementioned clustering algorithm is applied. From substantive and computational points of view, 'windows' $x(k) = \{x_{i_1 i_2}(k)\} \in R^{m \times n}$; $i_1 = 1, 2, \ldots, m$; $i_2 = 1, 2, \ldots, n$; $k = 1, 2, \ldots, N^* = MN(mm)^{-1}$ are more preferable for processing than vectorized data. Among the whole multitude, the best partition $P_p = \{Cl_1, Cl_2, \dots, Cl_p\}$ should be found in some accepted sense under condition of p non-overlapping clusters

$$Cl_l \cap Cl_q = \varnothing;\ l \neq q;\ l,\ q \in [1,\ p] \tag{1}$$

and matrix centroids $C = \{C(1),\ C(1), \dots,\ C(p)\}$, providing minimum value of the objective function

$$\min_{P_p} E = \min_{P_p} E(x(k),\ C(l)) = \min \sum_{k=1}^{N^*} \sum_{l=1}^{p} \mu(x(k),\ C(l)) Sp(x(k) - C(l))(x(k) - C(l))^T \tag{2}$$

where $Sp(\circ)$ denotes the trace of matrix, $\mu(x(k),\ C(l)) = \begin{cases} 1, \text{ if } x(k) \in Cl_l, \\ 0, \text{ otherwise.} \end{cases}$

Solution to the optimization problem of objective function (2) can be simply presented as follows:

$$C(l) = \frac{1}{N_l} \sum_{x(k) \in Cl_l} x(k) = \frac{\sum_{k=1}^{N^*} \mu(x(k),\ C(l)) x(k)}{\sum_{k=1}^{N^*} \mu(x(k),\ C(l))} \tag{3}$$

where N_l is the number of samples belonging to cluster Cl_l. In fact, the solution is reduced to finding matrix centers of weight in corresponding clusters.

Though, as it was already mentioned, condition (1) is not always held in real-world image processing applications, and cluster overlapping occurs very often. In other words, as a rule, it is impossible to set one-to-one correspondence of a ‘window’ with one or another particular segment. In vector space, such problems are successfully resolved with fuzzy cluster analysis where fuzzy C-means method (FCM) has gained the greatest popularity for pattern recognition and computer vision systems [1, 10-12].

FCM matrix modification was introduced in [13]. The following expression, there, was used as an objective function

$$E = E(x(k),\ C(l)) = \sum_{k=1}^{N^*} \sum_{l=1}^{p} \mu^{\beta}(x(k),\ C(l)) Sp(x(k) - C(l))(x(k) - C(l)) \tag{4}$$

with constraints

$$\sum_{l=1}^{p} \mu(x(k),\ C(l)) = 1\,;\ 0 < \sum_{l=1}^{m} \mu(x(k),\ C(l)) < N^*\,;\ l = 1, 2, \ldots, p\,. \quad (5)$$

Here, $\mu(x(k),\ C(l))$ is the value of sample $x(k)$ membership in cluster Cl_l, $\beta \geq 0$ is the parameter named 'fuzzifier' that specifies the level of edge overlapping for adjacent segments (usually $\beta = 2$).

The following result is obtained after optimization of objective function (4) under constraints (5):

$$\begin{cases} \mu(x(k),\ C(l)) = \dfrac{(Sp(x(k) - C(l))(x(k) - C(l))^T)^{\frac{1}{1-\beta}}}{\sum\limits_{l=1}^{p} (Sp(x(k) - C(l))(x(k) - C(l))^T)^{\frac{1}{1-\beta}}}, \\ C(l) = \dfrac{\sum\limits_{k=1}^{N^*} \mu^{\beta}(x(k),\ C(l))x(k)}{\sum\limits_{k=1}^{N^*} \mu^{\beta}(x(k),\ C(l))}, \end{cases} \quad (6)$$

and if $\beta = 2$, then the following simple expression is obtained [13] corresponding to popular J. Bezdek procedure generalization for the matrix case [14]:

$$\begin{cases} \mu(x(k),\ C(l)) = \dfrac{(Sp(x(k) - C(l))(x(k) - C(l))^T)^{-1}}{\sum\limits_{l=1}^{p} (Sp(x(k) - C(l))(x(k) - C(l))^T)^{-1}}, \\ C(l) = \dfrac{\sum\limits_{k=1}^{N^*} \mu^{2}(x(k),\ C(l))x(k)}{\sum\limits_{k=1}^{N^*} \mu^{2}(x(k),\ C(l))}. \end{cases} \quad (7)$$

2. Matrix Presentation of Fuzzy J-means

In connection with fuzzy image segmentation, fuzzy J-means method is reasonable to be extended to the matrix case. Moreover, it turns out that J-means significantly outperforms previously observed centroid based methods when many clusters and observations are present. Fuzzy J-means method consists in random movement of intermediary centroid (trapped into local extremum during optimization procedure) into empty points (that are not centroids) located in vicinity until reaching more 'deep' extremum being a centroid. Then, the obtained crisp solution is recalculated into fuzzy one by determining membership levels and refining all the cluster centroids.

FJM method is implemented in two main stages: finding local optimums using standard FCM with consecutive finding more 'deep' minimums via FJM-heuristic.

The first stage assumes the following sequence of steps.

Step 1. Specifying initial quite random partitioning $P_p = \{Cl_1, Cl_2, \dots, Cl_p\}$ with centroids $C(1)$, $C(2), \dots$, $C(p)$; fuzzifier β and threshold $\varepsilon > 0$ that determines condition of algorithm termination.

Step 2. Calculation of membership values $\mu(x(k), C(l))$ using the first relation from (6) (for arbitrary β) or the first relation from (7) (for $\beta = 2$) with centroids that are got from the previous step.

Step 3. Recalculation of centroids $C(1)$, $C(2), \dots$, $C(p)$ using the second relation from (6) (for arbitrary β) or the second relation from (7) (for $\beta = 2$) with membership values got from the previous step.

Step 4. Estimation of spherical norm of difference between the previously obtained centroids and those calculated at the third step.

Step 5. Check for termination conditions. If the obtained norm is less than ε, the algorithm terminates; if the obtained norm is greater than ε, return to step 2 with centroids located as after applying step 3.

The algorithm proceeds iteratively until termination condition is reached, and the solution will be presented by coordinates of local optimum in linear programming problem for (4), (5).

The second stage is a phase of jumps when random moves are performed in vicinity to the obtained local minimum in order to reach more 'deep' extremum.

It was shown in [10] that the general optimization problem (4) with constraints (5) can be reduced to unconstrained optimization with a specific presentation of objective function which can be written in matrix form as follows:

$$E = E(x(k), C(l)) = \sum_{k=1}^{N^*} \left(\sum_{l=1}^{p} (Sp(x(k) - C(l))(x(k) - C(l))^T)^{1-\beta} \right)^{1-\beta}, \qquad (8)$$

for arbitrary values of fuzzifier β and

$$E = E(x(k), C(l)) = \sum_{k=1}^{N^*} \left(\sum_{l=1}^{p} (Sp(x(k) - C(l))(x(k) - C(l))^T)^{-1} \right)^{-1} \qquad (9)$$

for $\beta = 2$.

Then, jumps are performed from any of the obtained centroids $C(l)$, $l = 1, 2...,$ p. In other words, the chosen centroid is replaced with any sample $x(r)$, and after that, the value of objective function (8) or (9) is calculated in the following view:

$$E = E(x(k),\ x(r)) = \sum_{k=1}^{N^*}(\sum_{l=1}^{p}(Sp(x(k)-x(r))(x(k)-x(r))^T)^{1-\beta})^{1-\beta} \qquad (10)$$

or

$$E = E(x(k),\ x(r)) = \sum_{k=1}^{N^*}(\sum_{l=1}^{p}(Sp(x(k)-x(r))(x(k)-x(r))^T)^{-1})^{-1}. \qquad (11)$$

If values (10), (11) turn out to be less than (8), (9) for some $x(r)$, the decision is made that a better centroid is found for cluster Cl_l. Then, all the membership values are recalculated using the first relation from (6) or (7). Such jumps are performed in vicinity of each centroid got at the first step. If it turns out that jumps in vicinity of all the centroids do not perfect the value of objective function (10), (11), then decision is made concerning termination of optimization process or jumps are performed again in vicinity of greater radius. Practically, this process may continue until empty points run to an end. The latest found local extremum is considered to be global one. Despite of seemingly bulky description, optimization process is quite simple from computational point of view.

Conclusion

It has been stated that strict boundaries between neighbouring segments are not always determinable in an image. For the purpose of quick and efficient image segmentation under condition of overlapping classes, matrix modification of fuzzy J-means method has been proposed. The developed method is based on matrix modification designed for fuzzy C-means clustering with the mechanism of random jumps, which provides finding global extremum of the accepted objective function. Computational simplicity of the method ensures its application for large-scale multidimensional data and guarantees resolving wide range of fuzzy clustering problems in matrix spaces.

BIBLIOGRAPHY

1. Ali, M.A. Review on fuzzy clustering algorithms / M.A. Ali, G.C. Karmakar, L.S. Dooley // IETECH Journal of Advanced Computations. – 2008. – Vol. 2, No. 3. – P. 169-181.

2. Gan, Y. Data Clustering: Theory, Algorithms and Applications / Y. Gan, Ch. Ma, J. Wu. – Philadelphia: SIAM, 2007. – 455 p.
3. Xu, R. Clustering / R. Xu, D.C. Wunsch. – Hoboken: Wiley, 2009. – 341 p.
4. Macqueen, J.B. Some methods for classification and analysis of multivariate observations / J.B. Macqueen // Mathematical Statistics and Probability: proc. of the 5th Berkeley symp., Berkley, 21 June - 18 July 1965 and 27 December 1965 - 7 January 1966. – Berkeley: University of California Press, 1967. – Vol. 1. – P. 281-297.
5. Hansen, P. J-Means: a new local search heuristic for minimum sum of squares clustering / P. Hansen, N. Mladenovic // Pattern Recognition. – 2001. – Vol. 34. – P. 405-413.
6. Rodriguez-Sotelo, J.L. An improved method for unsupervised analysis of ECG beats based on WT features and J-means clustering / J.L. Rodriguez-Sotelo, D. Cuesta-Frau, G. Castellanos-Dominguez // Computers in Cardiology. – 2007. – Vol. 34. – P. 581-584.
7. Xinye, Zh. Fast J-means algorithm: a constrained local search heuristic / Zh. Xinye, A. Jubai // Int. J. of Advancements in Computing Technology. – 2012. – Vol. 4. – P. 97-105.
8. Rastrigin, L.A. Statisticheskie metody poiska / L.A. Rastrigin. – M.: Nauka, 1968. – 376 p.
9. Belacel, N. Fuzzy J-means: a new heuristic for fuzzy clustering / N. Belacel, P. Hansen, N. Mladenovic // Pattern Recognition. – 2002. – Vol. 35, No. 10. – P. 2193-2200.
10. Hathaway, R.J. Optimization of clustering criteria by reformulation / R.J. Hathaway, J.C. Bezdek // IEEE Trans. on Fuzzy Systems. – 1995. – Vol. 3. – P. 241-245.
11. Pal, N.R. Sequential competitive learning and the fuzzy c-means clustering algorithm / N.R. Pal, J.C. Bezdek, R.J. Hathaway // Neural Networks. – 1996. – Vol. 9. – P. 787-796.
12. Bezdek, J.C. Fuzzy Models and Algorithms for Pattern Recognition and Image Processing / J.C. Bezdek, J. Keller, R. Krisnapuram, N.R. Pal. – N.Y.: Springer, 2005. – 776 p.
13. Bodyanskiy, Ye. Adaptive matrix fuzzy c-means clustering / Ye. Bodyanskiy, M. Skuratov, V. Volkova // East-west Fuzzy Colloquium 2012: proc. of the 19th Zittau fuzzy colloquium, Zittau, 5-7 Sep. 2012. – Zittau/Görlitz : Hochschule Zittau/Görlitz, 2012. – P. 111-116.

14. Bezdek, J.C. Pattern Recognition with Fuzzy Objective Function Algorithms / J.C. Bezdek. – N.Y.: Plenum Press, 1981. – 272 p.

 ISSN 1562-9945