UDK 004.932.2

S. Bogucharskiy, V. Mashtalir
Kharkiv National University of Radio Electronics,
Informatics department

ON MATRIX MODIFICATION OF CLARANS CLUSTERING METHOD IN LARGE VIDEO SURVEILLANCE DATABASES

Clustering algorithms for Very Large Data Bases (VLDB) are observed in application with image and video processing. Such a specific case requires initial data presentation as multidimensional vectors. That is why matrix modifications of traditional *k*-medoids, Partitioning Around Medoids, Clustering LARge Applications and CLARA based on RANdomized Search methods are proposed. Benefits and drawbacks of them all are examined.

Keywords – video processing, VLDB, *k*-medoids, PAM, CLARA, CLARANS

Introduction

Multidimensional data clustering aims at finding so called groups (classes, clusters, segments) of surveyed objects in information arrays being analyzed, which is an important part of data mining [1-5]. Its results are widely used for many applications. At the same time, there are quite a lot of practical tasks connected with processing of varied media, logical data, text and images, but the most popular and well-studied methods of cluster analyses become ineffective because of huge amounts of data needed to be processed.

Clustering algorithms for very large databases (VLDB) [1, 5], which form a separate branch in cluster analyses, turn out to be ahead of others for solving such kind of problems. One of the prime algorithms of such a type is CLARANS (Clustering Large Applications based on RANdomized Search) [6] based on well-known k-medoids method, PAM (Partitioning Around Medoids) [1] and CLARA (Clustering LARge Applications) algorithms, which are quite effective for processing not very large data arrays.

For the most part of existing clustering algorithms it should be noted that data to be processed are defined as multidimensional vectors forming traditional data table or multidimensional vector sequences (in case data are sequentially obtained for processing). At the same time, initial video processing information is usually presented in a matrix form, in which case this form may contain the whole image or its fragment. It is clear that in order to use some existing clustering methods for image processing, initial image should be vectorized somehow and devectorized to initial form after its processing.

The article presents CLARANS method modification that copes with initial data in a form of $(m\times n)$-matrix $x(k)=\{x_{i_1 i_2}(k)\}, i_1=1,2,...m; i_2=1,2,...n; k=1,2,...N$ is a number of observation in the analyzed data array. By doing so, it is considered that the whole of this array should be divided into p clusters, each of which may be described based on its centroid $c(l), l=1,2,..., p$ defined also in a form of $(m\times n)$-matrix.

Matrix modification of k-medoids method

As the classical k-means clustering method [5] is closely related to least squares criterion and Euclidean metric, k-medoids method is based on a matrix metric and least modules criterion which ensures its robustness to outliers. Suppose $x(k)$ and $w(l)$, then there exist $(n\times 1)$-vectors with the following distance between them:

$$D(x(k), c(L)) = \| x(k) - c(l) \|_1 = \sum_{i=1}^{n} | x_i(k) - c_i(l) |, \tag{1}$$

and the medoid is selected among available observations, for which the sum of distances within clusters (1) is minimal. It is not essentially hard to calculate the medoid-center, which components are the medians of corresponding components of each cluster elements.

At the same case, when $x(k)$ and $c(l)$ are $(m \times n)$-matrices, it is suggested to use the following distance instead of (1).

$$D(x(k), c(l)) = \sum_{i_1=1}^{m} \sum_{i_2=1}^{n} | x_{i_1 i_2}(k) - c_{i_1 i_2}(l) | = I_m^T | - x(k) - w(l) | I_n \tag{2}$$

where I_m, I_n are $(m \times 1)$- and $(n \times 1)$-vectors, formed by unities $| x(k) - w(l) | = \{ | x_{i_1 i_2}(k) - w_{i_1 i_2}(l) | \}$.

The process of finding medoids-centers can be implemented as follows:

i). Randomly select p observations $x(k)$ from data array and set them as initial medoids-centers $c_{(0)}(l), l = 1, 2, ..., p$.

ii). Assign the rest $N - p$ observations to the cluster with centroid that is nearest to each observation according to (2).

iii). In each cluster adjust all components of observations $x_{i_1 i_2}(k)$ in increasing order and find the median for each component $med\ x_{i_1 i_2}(k)$.

iv). Form a new centroid for each cluster $c_1(l) = \{ med\ x_{i_1 i_2}(k) \}$.

v). Repeat steps iii), iv) until

$$D(c_\tau(l), c_{\tau+1}(l)) \le \varepsilon \quad \forall l. \tag{3}$$

vi). After fulfillment of (3), set $c_{\tau+1}(l) = c(l)$ and calculate radius of each cluster as follows

$$R(l) = \arg \max_k D(c(l), x(k) \in c(l)). \tag{4}$$

The above method is efficient for not large N, but for very large data volumes enormous calculations for each observation in connection to the medoids reduce the method usage.

Matrix modification of partitioning around medoids

PAM method is more effective from computational point of view, as it does not calculate centroids, but sets existing observations to medoids of clusters.

To find p medoids, PAM starts from arbitrary selection of p image-matrices, much like it is done in k-medoids method. Further, at each iteration of the method, replacement of selected observation $x(k)$ is made to not yet selected image $x(q)$ from available sample set, until clusters will be eventually formed.

To estimate the replacement of $x(k)$ to $x(q)$, cost C_{rkq} is calculated at each iteration for all the rest of the data $x(r)$. In such a case, the cost is a distance function between observations

$$D(x(k), x(q)) = I_m^T | x(k) - x(q) | I_n. \tag{5}$$

During the replacement four different situations may occur.

i). Suppose $x(r)$ belongs to the cluster with medoid $x(k)$ at some iteration of the algorithm. Let $x(r)$ be closer by distance (5) to $x(t)$, than $x(q)$, i.e.

$$D(x(r), x(q)) \ge D(x(r), x(t)) \tag{6}$$

where $x(t)$ is the second medoid close to $x(r)$. Thus, if $x(k)$ is replaced to $x(q)$ as a medoid, $x(r)$ moves to cluster presented by $x(t)$. In such a case, the cost of replacement can be calculated as follows

$$C_{rkq} = D(x(r), x(t)) - D(x(r), x(k)), \tag{7}$$

and it is non-negative.

ii). Suppose $x(r)$ belongs to cluster presented by $x(k)$, with this $x(r)$ is ‘less common’ to $x(t)$ rather than to $x(q)$, i.e.

$$D(x(r), x(q)) < D(x(r), x(t)) . \tag{8}$$

Then, if $x(k)$ is replaced to $x(q)$, $x(r)$ will be related to cluster presented by $x(q)$. Thus, the cost of replacement equals to

$$C_{rkq} = D(x(r), x(q)) - D(x(r), x(k)) . \tag{9}$$

Unlike (7), it may be positive and negative as well, depending on image similarity to $x(r): x(k)$ or $x(q)$.

iii). Suppose $x(r)$ does not belong to cluster presented by $x(k)$, and $x(t)$ is a representative of this cluster. Further, let $x(r)$ be closer to $x(t)$ than to $x(q)$. Then, even if $x(k)$ is replaced to $x(q)$, $x(r)$ remains in cluster presented by $x(t)$ anyway. In this circumstances, the cost is

$$C_{rkq} = 0 . \tag{10}$$

iv). Suppose $x(r)$ belongs to cluster presented by $x(t)$, but $x(r)$ is less closer to $x(t)$ than to $x(q)$. Then, the replacement of $x(k)$ to $x(q)$ leads to transfer of $x(r)$ to cluster $x(q)$ from $x(t)$. In this circumstances, the cost equals

$$C_{rkq} = D(x(r), x(q)) - D(x(r), x(t)) \tag{11}$$

and it is always negative.

Combining all the four cases, it is clear that the total cost of the replacement of $x(k)$ to $x(q)$ is

$$TC_{kq} = \sum_r C_{rkq} . \tag{12}$$

This algorithm can be presented by the schema presented on Fig. 1.

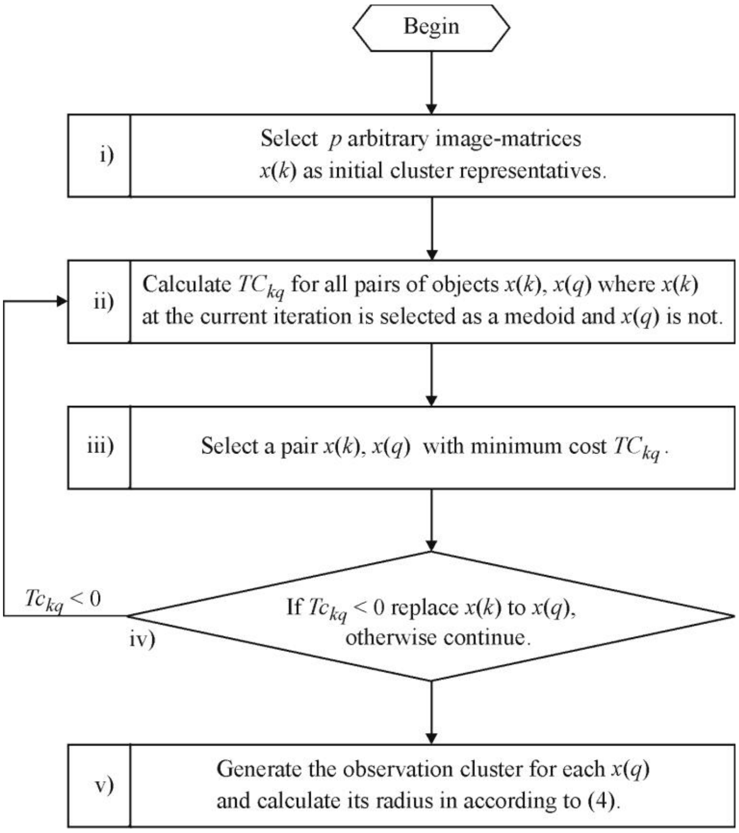

Begin
i)
Select p arbitrary image-matrices $x(k)$ as initial cluster representatives.
ii)
Calculate TC_{kq} for all pairs of objects $x(k)$, $x(q)$ where $x(k)$ at the current iteration is selected as a medoid and $x(q)$ is not.
iii)
Select a pair $x(k)$, $x(q)$ with minimum cost TC_{kq}.
$Tc_{kq} < 0$
iv)
If $Tc_{kq} < 0$ replace $x(k)$ to $x(q)$, otherwise continue.
v)
Generate the observation cluster for each $x(q)$ and calculate its radius in according to (4).

Fig.1. Method of partitioning around medoids

Although PAM method does not define medoids-centers, its implementation requires computations of distances, which makes it ineffective for quite large databases.

CLARA method

CLARA is an extension of PAM designed for large sample collections. Instead of searching for medoids of clusters in the whole sample set of N observations, this method finds medoids in each subsample that is much less by volume. For this purpose, several randomly selected subsamples are processed. The authors of this method have experimentally shown [7] that for the most part of tasks solved by them it was sufficient to form five subsamples containing $40+2p$ observations in each. This algorithm can be presented by the following schema (see Fig. 2).

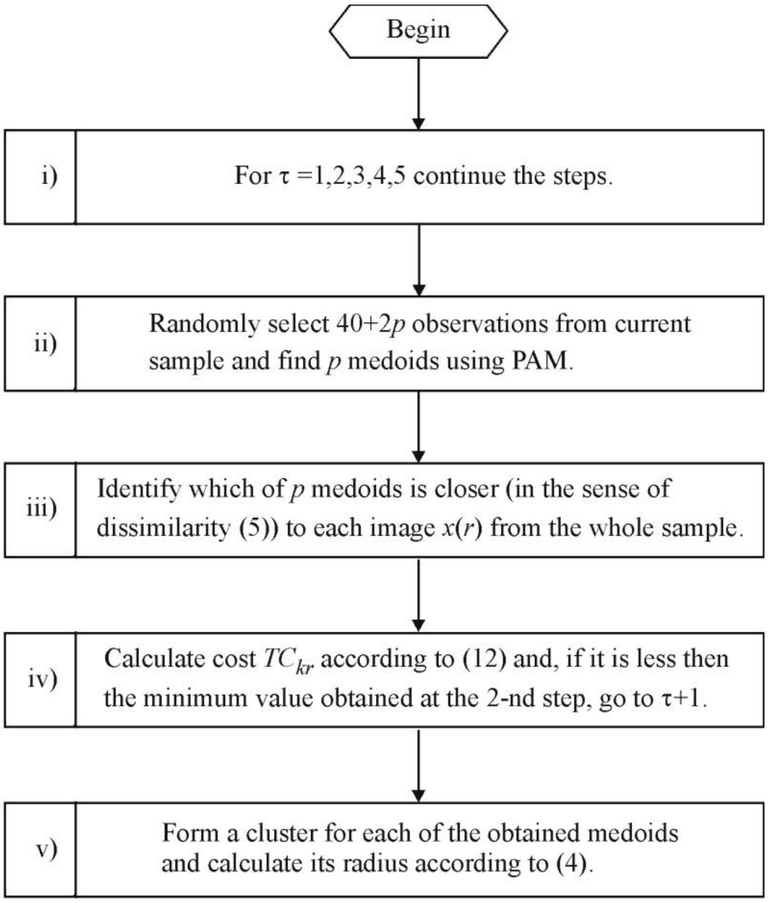

Begin
i)
For τ =1,2,3,4,5 continue the steps.
ii)
Randomly select 40+2*p* observations from current sample and find *p* medoids using PAM.
iii)
Identify which of *p* medoids is closer (in the sense of dissimilarity (5)) to each image *x*(*r*) from the whole sample.
iv)
Calculate cost TC_{kr} according to (12) and, if it is less then the minimum value obtained at the 2-nd step, go to τ+1.
v)
Form a cluster for each of the obtained medoids and calculate its radius according to (4).

Fig. 2. CLARA method

The authors of the method have shown its effectiveness for sample collections with nearly 1000 observations (that is not that much for video samples). Though there is a risk of loosing some information contained in analyzed observations.

CLARANS modification

CLARANS method is based on PAM and CLARA. Clustering process is treated as searching on the graph $G_{N,\,p}$, each of its nodes corresponds to one of p medoids. Cost is assigned for each of the nodes, which is defined by the distance between each of $N-p$ analysed images and current medoid values and calculated using equation (12).

It should be mentioned that earlier described PAM method also can be treated as searching $G_{N_1 p}$ graph minimum. With this, all the neighbours of current node are analyzed at each step (equations (5)-(12)), and then the current node is replaced by an observation that provide maximum decrease of the cost. Such process continues until global minimum is reached. Thus, at each iteration it is necessary to analyze $p(N-p)$ distances, which is too difficult for large N.

Apparently, CLARA analysis takes less images, and it is restricted to a subgraph of less dimensionality $G_{(40+2p),\,p} < G_{N,\,p}$, though, as it has been mentioned above, there is a risk of loosing data with important information.

CLARANS does not analyze each image at each node. Also it is not restricted to a subgraph only. For each node it randomly forms an area of local neighbourhood.

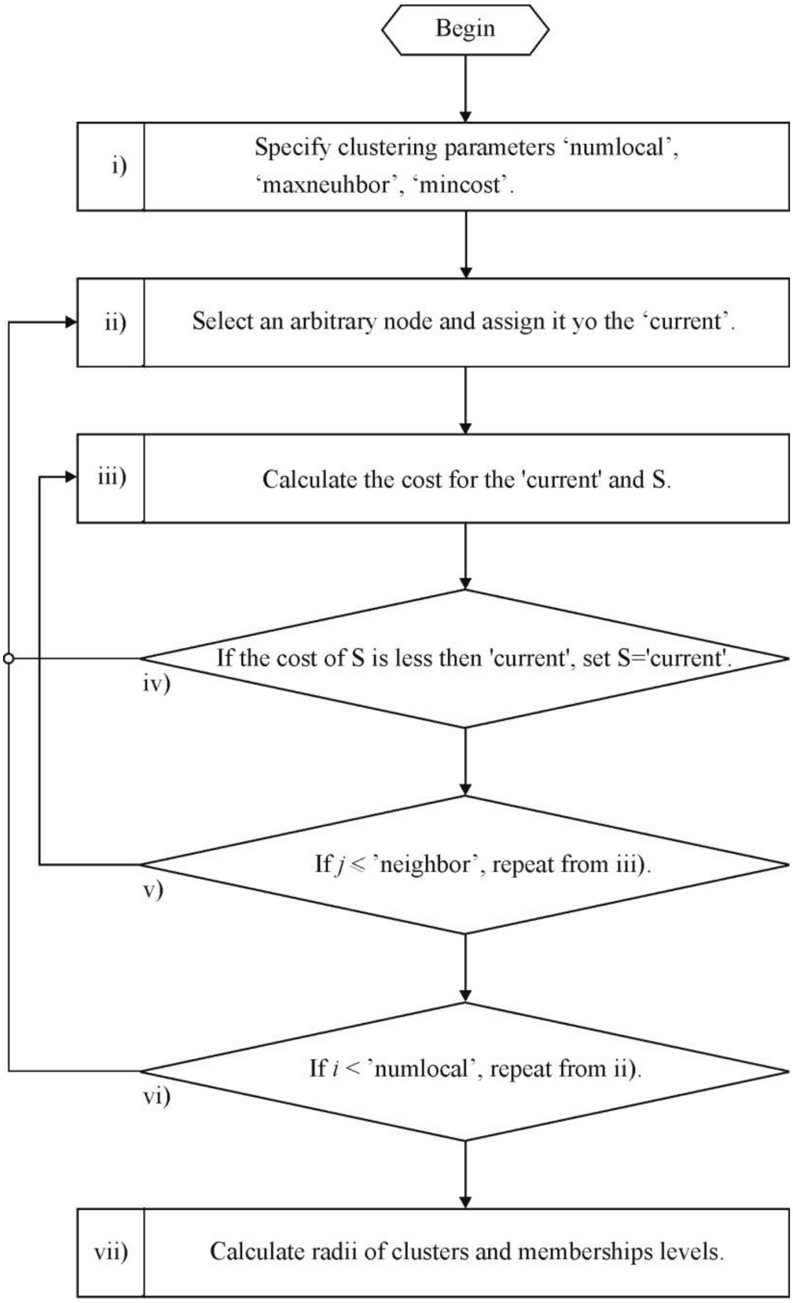
Clustering process based on modified CLARANS method can be implemented in the following sequence of steps.

i). Specify method parameters ‘numlocal’ and ‘maxneighbor’, assign $i=1$ and define ‘minconst’.

ii). Select arbitrary graph node $G_{N,\ p}$ and specify it as a ‘current’ one.

iii). Set $j=1$.

iv). Randomly select and image-neighbour S for the ‘current’ and according to equation (12) count costs for the ‘current’ and S.



Begin
i)
Specify clustering parameters ‘numlocal’, ‘maxneuhbor’, ‘mincost’.
ii)
Select an arbitrary node and assign it yo the ‘current’.
iii)
Calculate the cost for the 'current' and S.
If the cost of S is less then 'current', set S='current'.
iv)
If j < ’neighbor’, repeat from iii).
v)
If i < ’numlocal’, repeat from ii).
vi)
vii)
Calculate radii of clusters and memberships levels.

Fig.3. Modified CLARANS method

v). If the cost of S is lower, set S as ‘current’ and repeat from step iii).

vi). Otherwise increase j to $(j+1)$ and if $j \le 'neighbor'$, repeat from step iv).

vii). Otherwise if $j >$ ‘neighbor’, compare the cost of ‘current’ with ‘minconst’. If the cost of ‘current’<‘minconst’ assign ‘minconst’=‘current’.

viii). Increase i by 1 $(i=i+1)$, and if $i <$ ‘numlocal’ finish, otherwise repeat from step ii).

After $p =$ 'numlocal' clusters are formed, radius of each cluster can be calculated in addition according to equation (4). Moreover, as clusters almost always overlap in real-world applications (especially in image processing), levels of membership image to each cluster can be calculated. With this, if the distance between observation $x(k)$ and medoid $c(l)$ equals $D(x(k), c(l))$, then the level of belonging $x(k)$ to $c(l)$ can be defined using the equation

$$\mu(x(k), c(l)) = \frac{D^{-1}(x(k), c(l))}{\sum_{q=1}^{p} D^{-1}(x(k), c(q))}. \qquad (13)$$

Data processing algorithm based on modified CLARANS may be presented by the scheme shown on Fig. 3.

It is clear that clustering effectiveness depends greatly on the correct selection of parameters 'maxneighbor' and 'minconst', that requires users' qualification. It is also clear that the more 'maxneighbor' value is (under the limit $N - p = N -$ 'numlocal'), the more the process is alike to PAM and the more complicated it is. Thus, subjectivity level here is quite big, which is actually always present in tasks connected with self-learning paradigm, though it does not prevent finding acceptable solutions.

Conclusion

Modification of CLARANS method has been proposed, which is designed for clustering in very large databases (VLDB). The proposed modification copes with processing in a matrix form, and fuzzy clustering is applied for situations with overlapping clusters. The method is simple in implementation and does not need any complicated computations.

1. Han J., Kamber M. Data Mining: Concepts and Techniques. – 2-nd ed. – San Francisco: Morgan Kaufmann, 2006. – 800 p. 2. Gan G., Ma C., Wu J. Data Clustering: Theory, Algorithms, and Applications. – Philadelphia: SIAM, 2007. – 466 p. 3. Abonyi J., Feil B. Cluster Analysis for Data Mining and System Identification. – Basel: Birkhäuser, 2007. – 303 p. 4. Olson D.L., Dursun D. Advanced Data Mining Techniques. – Berlin: Springer, 2008. – 180 p. 5. Xu R., Wunsch D.C. Clustering. – Hoboken: John Wiley&Sons, 2008. – 358 p. 6. Kohonen T. Self-Organizing Maps. – 1-st ed. – Berlin: Springer, 1995. – 501 p. 7. Ng R.T., Han J. Efficient and clustering methods for spatial data mining // 20-th Int. Conf. on Very Large Data Bases. – Santiago de Chile, 1994. P.144-155. 8. Kaufman L., Rousseeuw P.J. Finding Groups in Data: An Introduction to Cluster Analysis. – N.Y.: John Wiley&Sons, 1990. – 342 p.