Image Anal Stereol 2023;42:133-144 doi: 105566/ias.2905
*Original Research Paper*

# GRADIENT DESCENT BATCH CLUSTERING FOR IMAGE CLASSIFICATION

JAE SAM PARK

Department of Electronics Engineering, Incheon National University, 119 Academy Road, Yeon Su Gu, Incheon, Korea.
e-mail: jaepark@inu.ac.kr

*(Received April 13, 2023; accepted June 16, 2023)*

ABSTRACT

The batch clustering algorithm for classification application requires the initial parameters and also has a drifting phenomenon for the stochastic process. The initial parameters are critical for the clustering to converge to the partial optimum. The drifting phenomenon in original batch clustering still has space to be improved thus to speed up the convergence based on the initial parameters. This paper proposes an unsupervised clustering method by addressing these two issues. Firstly, the estimation method for the initial parameters has been given in preliminary with a hierarchical manner of principal component analysis (PCA). The nonlinear parameters have been estimated based on a mathematical connection between PCA and clusters membership. With initial parameters, the drifting issue is addressed by combing the gradient descent and the batch clustering on an auxiliary objective to refine the initial parameters. The efficiency of the clustering process is proved based on the relationship between two quadratic functions followed by a justification. In addition, the effectiveness of the proposed method has been validated with the statistical F measure in classification application. The validation results show that the efficiency of the proposed gradient descent batch clustering has been improved significantly with trade-off to the accuracy in comparison of the original algorithms under the mean squared error (MSE) criterion.

Keywords: batch clustering, gradient descent, image classification, principal component analysis, stochastic process.

## INTRODUCTION

The real time classification of image data such as the natural citrus fruit image is essentially nontrivial when the data is posed in color space nonlinearly, *cf.* Jimenez *et al.* (2000). Even with the noise removal and the quality enhancement, the fundamental envelope spectra is generally a multiclass classification problem to be solved Li *et al.* (2012). Basically the classification methods have been formed in either supervised or unsupervised clustering, *cf.* Duda *et al*. (2000). Supervised classification methods are highly dependent on the credit of the training sample which is usually not available. Without ground truth, the unsupervised clustering methods are designed to find the hidden structure of the data by a competitive clustering using certain dissimilarity metrics without statistical model assumption. The clustering algorithms are broadly classified into three categories namely partitional, hierarchical, and density-based. Among which the hierarchical clustering discovers a sequence of partitions in a hierarchical structure represented by graphical dendrogram in agglomerative and divisive forms by merging the closest pair or splitting the farthest pair of objects to form clusters, *cf.* Xu and Wunsch(2005). The difficulty for hierarchical clustering algorithms is how to derive appropriate parameters for the termination condition. On the other hand, the density-based clustering with such as neighborhood and the number of points in region is not designed by locating the features especially when the data is strongly dense or overlapped Ester *et al*. (1996) . However, the partitional clustering incorporates the shape and the number of clusters by using certain metrics and prototypes. The well-known statistical c-means minimize the mean squared error (MSE) function as metrics for hyperspheres to classify the centroids of the clusters in form of the stochastic batch mode Linde *et al.* (2000) and the sample based gradient descent mode Macqueen (1967).

There are some topical issues for the clustering algorithms such as the local optimum and the initialization of the parameters. Based on the original C-means algorithms several attempts have been made to solve the local optimum issue such as by using GA (genetic algorithm) Krishna and Murty (1999), SA (simulated annealing) BandyoPadhyay (2001), and the hybrid of SA or

EA (evolutionary algorithm) Delport (1996). Also an enhanced Linde-Buzo-Gray (LBG) is developed using the concept of the utility of the code word to overcome the local optimum issue. However it is found that the estimation of the initial parameters is difficult to be addressed Patane, G. and M. Russo (2001). The original c-means have also been extended into other forms by incorporating with fuzzy logic membership function in the literature Du *et al.* (2006) and Xu *et al.* (2005) or combined with the static and active mechanism in *kd*-tree especially for large scale or higher dimensional data Lai *et al.* (2008) but with requirement of the initial parameters. More variants of the original algorithm have circumvented on the parameters and the local optimum issue using such as the constructive clustering technique ISODATA, self-creating learning algorithms, a method with imbalanced spatial distribution with a cluster, etc.

In literature, more extended works have been tried on metrics for example by combining different metrics Qian *et al.* (2016). Since the random initialization of the parameter for clustering method normally give poor classification results Chen *et al.* (2005), and an alternative initialization method has been proposed using the mean of the index in lower and upper area with consecutive Euclidean distance along one attribute Khan (2012). However the measure may not be constant along the direction for example a dominant axis with maximum variance for the optimal partitional direction Sujatha and Sona (2013). On the other hand, an spectral clustering algorithm has been proposed by selecting the most relevant eigenvector for analysis in clustering algorithm using Xiang and Gong (2008). With the maximum number of clusters given for the affinity matrix, the estimated weight parameter is sensitive to the dense data. In practice, the initialization of parameters becomes nontrivial when the data is nonlinear. Hence the better way to estimate is to use the variance information in the space of the data with such as PCA solution. On the other hand, the batch clustering process with a drifting issue still has space to be addressed even based on the original algorithms.

Basically two main clustering methods can be implemented such as the batch clustering and the gradient descent based methods. The gradient descent methods have three main variants. The stochastic gradient descent clustering update the parameter using each training example. The batch gradient descent updates the parameter for whole dataset which is convex uniquely. The mini-batch clustering using *n* samples to update the parameter. Even with the convergence of the stochastic batch clustering, two clustering algorithms perform differently with the initial parameters. In batch clustering, the pattern keeps changing from current cluster to the nearest cluster based on the update of the parameters with the simple competitive rule. Hence the update of mean of cluster with batch clustering still drifts around the ideal centroids. On the other hand, the gradient descent clustering variants update the parameters with simple sample online, fixed number of samples, or the whole data samples for such as regression. However, the variants have not combined the gradient descent with the batch clustering on the multivariate non-convex case, meanwhile the number of samples belong to different nonlinear clusters with a parameter of centroid. Basically the optimization variants for the gradient descent have focused on the learning rate, the acceleration of SGD process with momentum, and the other strategies on top of the distributing methods such as data shuffling or by using curriculum learning Zaremba and Sutskever (2015) or batch normalization Ioffe and Szegedy (2015). Since the data such as the natural citrus color image data are normally non-convex in color space, the number of parameters for various color clusters is a nonlinear problem. If the gradient descent can be applied on each of clusters, the parameters of centroids can be directed to the minimum in negative gradient direction thus to address the drifting phenomenon. To apply the gradient descent on the nonlinear parameters, an auxiliary objective need to be reformed based on the original one. In addition, the clustering process still requires the initial parameters in preliminary.

The contents are organized as follows. In section 2, the initialization of parameters is given in preliminary with a mathematical connection between PCA solution and the clustering membership. To refine the initialized parameters, the gradient descent batch clustering is derived from the reformed objective. The speed-up of the clustering convergence is proved using the relationship between the quadratic functions followed by a mathematical justification. The method is validated with statistical measure in comparison of the other original algorithms. The last section draws a conclusion

## MATERIALS AND METHODS

In this section, a gradient descent batch clustering algorithm is proposed for the nonlinear classification application. The proposed method is based on the preliminary for the initialization of parameters namely the centroids of the clusters. These parameters are refined to the partial optimum by the modified batch clustering algorithm.

When the data is posed in certain color space nonlinearly, for example the natural citrus fruit color image,

it is difficult to reveal the structure of the clusters in any one eigenvector subspace. Hence an iterative application of PCA solution in a hierarchical manner is proposed to reveal the hidden clusters in different subspaces of subsets. Since PCA solution is obtained by maximization of variance of data in eigenvector subspace, the optimal solution has been found to indicate the membership of at least two subsets in the subspace. The statistical PCA which is known as Karhunen-Loeve transform (KLT) Rao and Yip (2001) has a mapping function to extract a certain feature in a low-dimensional space. PCA automatically extracts the eigenvectors based on the maximization of the variance of the projected data using the centered covariance matrix of the data Jolliffe (2002). The following defines PCA in the image data.

**Definition 2.1** Consider an original image data with *p* random variables arranged as a column vector, denoted **x**. The *k*th PC (principal component), $y_k$, where $k$=1, 2,…, $p$, is given by:

$$y_k = \mathbf{w}_k^T \mathbf{x} \tag{1}$$

where $\mathbf{w}_k = [w_1, w_2, \cdots, w_p]^T$ is an eigenvector of the covariance matrix corresponding to its eigenvalue $\lambda_k$. The main objective of PCA is to find $\mathbf{w}_k$ providing the maximum variance of the *k*th PC using the second central moment as presented in Eq.2:

$$\text{var}\left(\mathbf{w}_k^T \mathbf{x}\right) = E\left[\left(\mathbf{w}_k^T \mathbf{x} - E\left(\mathbf{w}_k^T \mathbf{x}\right)\right)^2\right] = \mathbf{w}_k^T \boldsymbol{\Sigma} \mathbf{w}_k \tag{2}$$

where $E\left(\mathbf{w}_k^T \mathbf{x}\right)$ is the expectation of the *k*th PC variable and var(.) represents the variance and $\boldsymbol{\Sigma}$ is the covariance matrix of the elements of vector **x**. The result of Eq.2 is the objective function to be maximized to find the largest PC. Since the maximum will not be determined for finite $\mathbf{w}_k$, a normalized constraint $\mathbf{w}_k^T \mathbf{w}_k = 1$ must be imposed. The quadratic objective function can be rewritten as in Eq.3:

$$\text{Maximize: } \mathbf{w}_k^T \boldsymbol{\Sigma} \mathbf{w}_k \tag{3}$$

$$\text{Subjective to: } \mathbf{w}_k^T \mathbf{w}_k = 1$$

By combining the normalization constraint to the objective function, the standard optimization problem becomes Eq.4 using the techniques of Lagrange Multipliers :

$$\text{Maximize: } \mathbf{w}_k^T \boldsymbol{\Sigma} \mathbf{w}_k - \lambda_k \left(\mathbf{w}_k^T \mathbf{w}_k - 1\right) \tag{4}$$

where $\lambda_k$ is a Lagrange multiplier. Taking a derivation of Eq.4 with respect to the variable $\mathbf{w}_k$ gives the standard characteristic function:

$$\boldsymbol{\Sigma} \mathbf{w}_k - \lambda_k \mathbf{w}_k = 0 \tag{5}$$

This function can be expressed in another form by $\left(\boldsymbol{\Sigma} - \lambda_k \mathbf{I}_p\right)\mathbf{w}_k = 0$, where $\mathbf{I}_p$ is an $(p \times p)$ identity matrix. Here $\lambda_k$ is the eigenvalue, and $\mathbf{w}_k$ is the corresponding eigenvector of covariance matrix $\boldsymbol{\Sigma}$. Substituting Eq.5 into the objective function, and using the normalization constraint, the solution for the maximum variance of the variable is the eigenvector corresponding to the largest eigenvalue of the covariance matrix $\boldsymbol{\Sigma}$ as shown in Eq.6:

$$\mathbf{w}_k^T \boldsymbol{\Sigma} \mathbf{w}_k = \mathbf{w}_k^T \lambda_k \mathbf{w}_k = \lambda_k \mathbf{w}_k^T \mathbf{w}_k = \lambda_k \tag{6}$$

PCA solution is to find the eigenvector $\mathbf{w}_k$ and the corresponding eigenvalue $\lambda_k$, which provide the maximum variance of vector **x** in *k*th PC direction. When there are number of nonlinear subsets or classes, at least the distance between two main subsets is maximized in the maximal eigenvector direction. PCA solution for the subsets is also independent from each other when the subsets are not interrelated. The use of PCA solution is based on a connection between the PCA solution and the membership of clusters. Firstly, the principal components from PCA solution indicates the membership of clusters. On top of that the maximization of distance between two subsets always exist independently for two subsets. The mathematical verification and theorems follow the works Otsu (1979), Ding and He (2004). The definition of C-means is then given by Eq.7 using the number of *K* centroids to cluster the data:

$$J_K = \sum_{k=1}^{K} \sum_{i \in C_k} (\mathbf{x}_i - \mathbf{c}_k)^2 \tag{7}$$

where $\{\mathbf{x}_i : \mathbf{x}_i \in C\}$, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n]^T$ is the data matrix for data set $C$, $\mathbf{c}_k = \sum_{i \in C_k} \mathbf{x}_i / n_k$ is the centroid of the subset $C_k$, $n_k$ is the size of subset $C_k$ and $\sum_{k=1}^{K} n_k = n$, $K$ is the total number of all subsets, $\{C_k : C_k \in C\}$ and $C = C_1 \oplus C_2 ..... \oplus C_K$.

Let

$$d(C_k, C_l) = \sum_{i \in C_k} \sum_{j \in C_l} (\mathbf{x}_i - \mathbf{x}_j)^2 \tag{8}$$

be the sum of squared distance (SSD) between two subsets $C_k$ and $C_l$. When $k = l$, SSD between individuals within one class becomes the deformation formula of covariance for one subset as in Eq.9. The proof refers to Zhang *et al.* (2012).

$$\text{var}(C_k, C_k) = \frac{1}{2n_k^2} \sum_{i=1}^{n_k} \sum_{j=1}^{n_k} (\mathbf{x}_i - \mathbf{x}_j)^2 \tag{9}$$

In this research, since the image data is divided into 2 main subsets, the case with $K = 2$ is considered to show the connection between PCA solution and the

membership of clusters in C-means algorithm. Note that this could be easily extended for more general cases. Denote $C_1$ and $C_2$ as two subsets, where $C_1 \oplus C_2 = C$. Also note that $n_1/n + n_2/n = 1$ where $n_1/n$ and $n_2/n$ are the class occurrence probabilities. It is obvious from the definition that SSD of the whole data set $C$ can be calculated using SSD within each of two subsets and SSD between two subsets as in Eq.10:

$$d(C,C) = d(C_1,C_1) + d(C_2,C_2) + 2d(C_1,C_2) \quad (10)$$

Substituting Eq.8 into Eq.9, the covariance of the whole data set $C$ can be expressed in Eq.11:

$$\sigma_T^2 = \overline{\mathbf{y}^2} = \sum_i \mathbf{y}_i^T \mathbf{y}_i / n = \frac{1}{2n^2} d(C,C) \quad (11)$$

where $\sigma_T^2$ is the variance for the whole data set $C$, $\mathbf{y}_i = \mathbf{x}_i - \bar{\mathbf{x}}$ and $\bar{\mathbf{x}} = \sum_i \mathbf{x}_i / n$. With Eq.11, Eq.10 can be expressed by Eq.12:

$$2n^2\sigma_T^2 = d(C_1,C_1) + d(C_2,C_2) + 2d(C_1,C_2) \quad (12)$$

Since the objective function $J_K$ for C-means clustering is related to the within class variance for all subsets based on the definition, using Eq.9, $J_K$ can be expressed by Eq.13:

$$J_K = \sum_{k=1}^K \sum_{i \in C_k} (x_i - c_k)^2 = \sum_{k=1}^K n_k \sigma_k^2 = \sum_{k=1}^K \frac{1}{2n_k} \sum_{i,j \in C_k} (x_i - x_j)^2 = \frac{1}{2n_1} d(C_1,C_1) + \frac{1}{2n_2} d(C_2,C_2) \quad (13)$$

where $\sigma_k^2$ is the within class variance for the subset $C_k$. Some algebra yields the following Eq.14 from Eq.12:

$$n\sigma_T^2 = \frac{1}{2n_1}(1 - \frac{n_2}{n}) d(C_1,C_1) + \frac{1}{2n_2}(1 - \frac{n_1}{n}) d(C_2,C_2) + \frac{1}{n} d(C_1,C_2) \quad (14)$$

After rearrangement of Eq.14, the objective function $J_K$ can be re-expressed as Eq.15:

$$J_K = n\sigma_T^2 - \frac{n_1 n_2}{2n} (\frac{2d(C_1,C_2)}{n_1 n_2} - \frac{d(C_1,C_1)}{n_1^2} - \frac{d(C_2,C_2)}{n_2^2}) = n\sigma_T^2 - \frac{1}{2} J_D \quad (15)$$

where the right part is denoted as distance objective function, $J_D = \frac{n_1 n_2}{2n} (\frac{2d(C_1,C_2)}{n_1 n_2} - \frac{d(C_1,C_1)}{n_1^2} - \frac{d(C_2,C_2)}{n_2^2})$.

On the other hand, the following relation always holds Otsu (1979):

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2 \quad (16)$$

where $\sigma_W^2$ is the within class variance for two subsets, $\sigma_W^2 = \mu_1\sigma_1^2 + \mu_2\sigma_2^2$, and $\mu_1$ and $\mu_2$ are the class occurrence probability, here $\mu_1 = n_1/n$ and $\mu_2 = n_2/n$. The variables $\sigma_1$ and $\sigma_2$ are the variance of subset $C_1$ and $C_2$, and $\sigma_B^2 = \mu_1\mu_2(\mathbf{c}_1 - \mathbf{c}_2)^2$ is the between class variance, where $\mathbf{c}_1$ and $\mathbf{c}_2$ are the centroids of two subsets. By substituting $\sigma_T^2$, the total variance from Eq.16 into Eq.12 using Eq.9 for within class variance, it can be found that SSD between two subsets can be expressed as in Eq.17:

$$\frac{1}{n_1 n_2} d(C_1,C_2) = \frac{1}{2n_1^2} d(C_1,C_1) + \frac{1}{2n_2^2} d(C_2,C_2) + (c_1 - c_2)^2 \quad (17)$$

By substituting Eq.17 into $J_D$ in Eq.15, $J_D$ becomes Eq.18:

$$J_D = \frac{n_1 n_2}{n} (\mathbf{c}_1 - \mathbf{c}_2)^2 \quad (18)$$

Eq.15 and Eq.18 indicate that the minimization of $J_K$ is equivalent to the maximization of SSD between two subsets in the distance objective function $J_D$ which is always positive, as given in Theorem 2.1.

**Theorem** 2.1 For $K$ = 2, the minimization of C-means cluster objective function $J_K$ is equivalent to the maximization of the distance objective $J_D$, which is always positive.

Proof. See Ding (2004). □

Next thing is to show that the eigenvector subspace indicates the membership for C-means clusters. Before that, the theorem for singular value decomposition (SVD) is necessary. With data matrix $\mathbf{X}$, let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n]^T$ be the centered data matrix, where $\mathbf{y}_i = \mathbf{x}_i - \bar{\mathbf{x}}$ and $\bar{\mathbf{x}} = \sum_i \mathbf{x}_i / n$. Then $\mathbf{A} = \sum_{ij} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})^T$ is the covariance matrix which can be expressed by $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$, where $\mathbf{U}$ and $\mathbf{V}$ are two orthogonal matrices, with $rank(\mathbf{A}) = r$. Hence $\mathbf{A}$ is expressed by $\mathbf{A} = \sum_i \lambda_i \mathbf{u}_i \mathbf{v}_i^T$, where $\lambda_i$ are singular values, $\mathbf{u}_i$ and $\mathbf{v}_i$ are principal directions and principal components respectively [62]. The following theorem shows the connection between the clusters membership indicator and PCA solution.

**Theorem** 2.2 For C-means clustering where $K$ = 2, the continuous solution of the cluster indicator vector is the principal component $\mathbf{v}_1$, i.e., clusters $C_1$, $C_2$ are given by

$$C_1 = \{i : \mathbf{v}_1(i) \le 0\}, \quad C_2 = \{i : \mathbf{v}_2(i) > 0\} \quad (19)$$

The optimal value of C-means objective satisfies the bounds

$$n\overline{\mathbf{y}^2} - \lambda_1 < J_{K=2} < n\overline{\mathbf{y}^2} \quad (20)$$

Proof. See Ding (2004). □

The covariance matrix is constructed by the sum of centered data for each variable. Then only the case with $K = 2$ is considered with meaning to classify the data into two main subsets in the proposed method. Note that this can be extended to more general case with $K > 2$. The existence of maximization between two independent subsets is given as follows.

**Theorem** 2.3 For $K = 2$ case with the probability of class occurrence for two data subsets, the maximization of the between class distance always exist.

Proof. The maximization of SSD between two subsets in the distance objective $J_D$ in Eq.18 is equivalent to the maximization of $\sigma_B^2 = \mu_1\mu_2(\mathbf{c}_1 - \mathbf{c}_2)^2$ . Hence the range of thresholding $t$ is sought to maximize:

$$T = \{t : \mu_1\mu_2 = \mu(t)[1-\mu(t)]\} \quad (21)$$

There are two situations only. If all the data belong to one class, then factor $\mu(t)$ is zero or one which means no subsets in the data set. Otherwise $0 < \mu(t) < 1$ is true, then $0 < 1-\mu(t) < 1$ , hence $0 < \mu(t)[1-\mu(t)] < 1$ is true. Therefore, the maximum always exists. □

Theorem 2.1 and theorem 2.2 have shown the connection between PCA solution and the clusters' membership for C-means algorithm. The indication for minimum two subsets with maximization of between class distances is guaranteed. Another consideration is the termination of the use of PCA solution. Under the condition that **x** has a normal distribution, i.e., the ellipsoids given in Eq.22 which define contours of constant probability for the distribution of **x**, the following property shows that PCs of such ellipsoids represent the axes which can provide the maximal statistical variation and those axes are orthogonal from each other.

**Property 2.1** Considering the $p$-dimensional ellipsoids

$$\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x} = constant \quad (22)$$

then the PCs define the principal axes of these ellipsoids. Here **x** is the variable and **Σ** is the covariance matrix of the elements of vector **x**.

Proof. See Jolliffe (2002). □

The geometric property of ellipsoids is of statistical relevance if the distribution of **x** is assumed as multivariate normal. The thresholds on the projection data on the maximal eigenvector generally become less credible as the number of classes to be separated increases nonlinearly. However, PCs can be used to suggest suitable two main subsets based on the variance of intra classes from the mathematical connection. Then the global threshold on the projection data can be used to segment the data into two members in the maximal eigenvector subspace. This process is applied on the segmented or available subsets iteratively to reveal the hidden clusters hierarchically as given in table 1. In classification application, the process starts with the rearrangement of the image data as a ( $n \times m$ by 3) matrix using for example a*, b* components from CIE color space Lab and Hue component from HSV color space where $n \times m$ is the resolution of image.

Table 1. *Algorithm 1 for initial parameters estimation with PCA solution.*

Input 1: Image data.

Input 2: Criterion of confidence level for variance of PCs

Output 1: Initial mean of clusters

Condition 1: Standard deviation of PCs<Criterion

Start

1. Rearrange image data according to a* b* and Hue from CIE color space Lab and HSV respectively

2. Apply PCA to find eigenvectors and eigenvalues for each (segmented) image data

3. Check the termination criteria

4. If the criteria is not met

5. Project the image onto the maximum eigenvector space

6. Apply a global thresholding providing the maximum variance in the subspace to divide each image data into two segmented images Otsu (1979)

7. Else

End

With initial parameters given without disparity to the simplified objective in previous sections, the competitive learning is a statistical cluster analysis which partitions $n$ input patterns data, **X**, into $K$ separated subsets by minimizing the mean squared error in Eq.23:

$$E = \frac{1}{n_k}\sum_{p=1}^{n_k}\sum_{k=1}^{K} w_{kp}\left\|\mathbf{x}_p - \mathbf{c}_k\right\|^2 \tag{23}$$

where $K$ is the number of centroids of clusters, $n_k$ is the number of data in cluster $C_k$, $w_{\mathrm{kp}}$ is the connection weight assigned to prototype $C_k$ with respect to $\mathbf{x}_p$, denoting the membership of data $p$ into cluster $k$. It is difficult to use the gradient descent method, since the winning prototypes must be determined with respect to each input pattern $\mathbf{x}_p$. By using the following functional, the gradient-descent method leads to sequential updating of the prototypes with respect to the input pattern $\mathbf{x}_p$.

$$E = \sum_{k=1}^{K} w_{kp}\left\|\mathbf{x}_p - \mathbf{c}_k\right\|^2 \tag{24}$$

Based on Eq.24, assuming $\mathbf{c}_w = \mathbf{c}_w(t)$ to be the winning cluster centroid for input data $\mathbf{x} = \mathbf{x}_t$ at time $t$. The gradient descent method leads to the sequential update of the prototype (centroid of cluster $k$) as follows.

$$\mathbf{c}_w(t+1) = \mathbf{c}_w(t) + \eta(t)\left[\mathbf{x}_t - \mathbf{c}_w(t)\right] \tag{25}$$

$$\mathbf{c}_i(t+1) = \mathbf{c}_i(t)\ ,\ i \neq w \tag{26}$$

The gradient descent performs the update of parameter upon each example presented which causes the objective oscillate.

On the other hand, the batch clustering is derived by minimizing the metric between the data and the relative clusters which it is close to in Eq.27.

$$\text{Minimize: } f(\mathbf{X},\mathbf{C}) = \sum_{i=1}^{n}\sum_{j=1}^{k} w_{ij} D\left(\mathbf{x}_i, \mathbf{c}_j\right) \tag{27}$$

$$\text{Subject to: } \sum_{j=1}^{k} w_{ij} = 1,\quad j = 1,...,k$$

$$w_{ij} = 0 \quad or \quad 1,\quad i = 1,...n,\quad j = 1,...,k$$

where $D$ is certain metric for competitive rule, $\mathbf{x}_i$ is one data instance, $\mathbf{c}_j$ is mean of one data cluster. Assigning the instance to clusters is fixing the probability parameter $w_{ij}$, and then Eq.27 can be expressed in the mean squared error (MSE) function in Eq.28:

$$E(\mathbf{c}_1,...,\mathbf{c}_k) = \frac{1}{n}\sum_{j=1}^{k}\sum_{\mathbf{x}_i \in \mathbf{c}_j}\left\|\mathbf{x}_i - \mathbf{c}_j\right\|^2 \tag{28}$$

where Euclidean metrics is measured as the competitive rule to find the closest cluster $\mathbf{c}_j$ for the instance $\mathbf{x}_i$. By minimizing $E$ with respect to the centroids $\mathbf{c}_j$ and by setting the derivative $\partial E/\partial \mathbf{c}_j$ to zero, the original batch c-means is obtained in terms of time $t$ in Eq.29:

$$\mathbf{c}(t+1)_j = \frac{1}{n_j}\sum_{\mathbf{x}_i \in \mathbf{c}_j} \mathbf{x}(t)_i \tag{29}$$

where $n_j$ is the size of the instance in each cluster. At each step, the patterns keep changing from one cluster to the nearest cluster with $j = \arg_j \min\left\|\mathbf{x} - \mathbf{c}_j\right\|$. The gradient descent clustering continues to update the prototype upon the presentation of each new data pattern using in Eq.3 which is inefficient even with the random initialization of the prototypes or ad hoc approaches. On the other hand, the batch clustering still has a drifting issue using Eq.29 [13] for update of the prototype stochastically based on the competitive rule. Since the stochastic process doesn't use search direction to certain minimum of parameter, the way to solve the drifting issue is to minimize the objective by directing the solution to the minimum in search line of such as gradient descent manner. Hence to apply the gradient descent on batch clustering, an auxiliary objective is necessary which includes a dependent variable such as the mean of each cluster and one input pattern by using the mean of cluster based on the original batch rule.

## PROPOSAL OF OBJECTIVE FOR GRADIENT DESCENT BATCH CLUSTERING METHOD

To propose the auxiliary objective, consider any two vectors $\mathbf{x}$ and $\mathbf{y}$ in $E^n$, the Cauchy-Schwarz inequality holds: $\left|\langle \mathbf{x}, \mathbf{y}\rangle\right| \le \|\mathbf{x}\|\|\mathbf{y}\|$. As for case in Euclidean space, the triangle inequality follows the Cauchy-Schwarz inequality which holds as follow $\|\mathbf{x}+\mathbf{y}\|^2 \le \|\mathbf{x}\|^2 + 2\|\mathbf{x}\|\|\mathbf{y}\| + \|\mathbf{y}\|^2$. Assuming there are $k$ clusters in the data set $C$, each data $\mathbf{x}_i$ is covered by at least one of the clusters. Let any two vectors $\mathbf{x}$ and $\mathbf{y}$ to be replaced by the error between data $\mathbf{x}_i$ and the mean of the winning cluster by $\mathbf{x}_i - \mathbf{c}_j$, for all $\mathbf{x}_i \in C_j$, $\mathbf{c}_j$ is the mean of cluster $j$ with number of $n_j$ data, and $j = 1,...,k$, $k$ is the number of clusters in data set $C$. The two data samples can be generalized for $n$ number of data. Each of data wins at least one of clusters with simple competitive rule, then the squared total error holds as follows:

$$\left\|\sum_{j=1}^{k}\sum_{x_i \in C_j}(x_i - c_j)\right\|^2 \le \sum_{j=1}^{k}\sum_{x_i \in C_j}\|x_i - c_j\|^2 + 2\sum_{j,m=1, j\neq m}^{k}\sum_{x_i \in C_j, x_l \in C_m}\|x_i - c_j\|\|x_l - c_m\| \tag{30}$$

The first part of the right hand side is the sum of squared errors. The second part is the product of norm of two error factors which is no less than zero. Each of

two error factors is minimized using the simple competitive rule. Hence the minimization of the squared total error is proportional to the sum of squared errors. The squared total error can be formed into the sum of errors for each different clusters as follows:

$$\left\| \sum_{j=1}^{k} \sum_{\mathbf{x}_i \in C_j} \left( \mathbf{x}_i - \mathbf{c}_j \right) \right\|^2 = \left\| \sum_{j=1}^{k} \left( \sum_{i=1}^{n_j} \mathbf{x}_i - n_j \mathbf{c}_j \right) \right\|^2 \quad (31)$$

Now let the sum of each cluster data in Eq.31 as dependent variable of independent data. The mean of the cluster is determined with respect to all the winning data to the cluster. Since the mean of each cluster is unique, the objective is to find the dependent variable to be close to the ideal mean of the cluster by including all the winning data instances into the cluster. Since the parameters of centroids of clusters have been estimated in previous section, then the gradient descent can be applied to the deformed objective function with respect to each variable of mean of cluster by leaving others as constant in Eq.31 following the definition of the gradient descent (Defined as steepest descent).

**Definition 2.2** The method of steepest descent is defined by the iterative algorithm

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha_t \mathbf{g}_t \quad (32)$$

where $\alpha_t$ is a nonnegative scalar minimizing $f\left(\mathbf{x}_t - \alpha_t \mathbf{g}_t\right)$ and the gradient $\mathbf{g}(\mathbf{x}_t) = \nabla f\left(\mathbf{x}_t\right)^T$ is defined as column vector. The search is along the negative direction of gradient to the minimum. The gradient descent for the objective is as follows.

$$\text{Minimize: } E = \frac{1}{n} \sum_{j=1}^{k} \left\| \mathbf{s}_j - n_j \mathbf{c}_j \right\|^2 \quad (33)$$

$\mathbf{s}_j = \sum_{i=1}^{n_j} \mathbf{x}_i$ , $\mathbf{x}_i \in C_j$ with $\|\mathbf{x}_i - \mathbf{c}_k\| = \min_j \|\mathbf{x}_i - \mathbf{c}_j\|$ and $\sum_{j=1}^{k} w_{ij} = 1, \quad j = 1,...,k$ , here $w_{ij} = 0 \quad or \quad 1, \quad i = 1,...n, \quad j = 1,...,k$ , where $n_j$ is the number of all winning data to cluster $C_j$, $k$ is the number of clusters in data set $C$, and $n$ is the total number of data. The probability $w_{ij}$ is fixed by assigning all the $n_j$ winning data in one cluster. The first order derivative with respect to variable $\mathbf{c}_j$ is given by $\frac{\partial E}{\partial \mathbf{c}_j} = -\frac{2n_j}{n}\left(\mathbf{s}_j - n_j \mathbf{c}_j\right)$. Since the gradient of the objective with respect to the variable $\mathbf{c}_j$ vanishes at a relative local minimum of each mean of clusters, and further the multiple of vector by a constant doesn't change the direction of the vector, the gradient can be presented by $\frac{\partial E}{\partial \mathbf{c}_j} = -\frac{2n_j^2}{n}\left(\frac{\mathbf{s}_j}{n_j} - \mathbf{c}_j\right)$, by pertaining to the same relative local minimum vector. Hence by including the constants of the gradient into the nonnegative scalar $\eta(t)$, the gradient descent batch rule for Eq.33 is given by Eq.34.

$$\mathbf{c}_j(t+1) = \mathbf{c}_j(t) + \eta(t)\left[\frac{1}{n_j} \sum_{\mathbf{x}(t)_i \in \mathbf{c}_j} \mathbf{x}(t)_i - \mathbf{c}_j(t)\right] \quad (34)$$

where $\left[\frac{1}{n_j} \sum_{\mathbf{x}(t)_i \in \mathbf{c}_j} \mathbf{x}(t)_i - \mathbf{c}_j(t)\right]$ decides the direction of the negative gradient to the minimum of the parameter. The objective is unconstrained since the sum of the winning data in each cluster is changed based on the simple competitive rule. Hence the gradient descent batch clustering is based on the update of the mean of cluster and the initial parameters. Since the objective has a similar quadratic form to the original objective, the convergence is justified by relating to the original quadratic form as follows. □

The part of gradient can be interpreted as the distortion between the centroid to be converged to and the mean of cluster in a mathematical justification. Following the definition and assigning the instance to clusters, the original c-means batch clustering by Eq.29 can be expressed in Eq.34 without changing the arithmetic by applying the clustering parameter $\eta_1(t)$ and denotes:

$$\Delta_1 = \eta_1(t)\left[\frac{1}{n_j} \sum_{\mathbf{x}_i(t) \in \mathbf{c}_j} \mathbf{x}_i(t) - \mathbf{c}_j(t)\right] \quad (35)$$

where $\eta_1(t) = 1$ to maintain the original starting point, $\Delta_1$ in Eq.35 is the distortion between the mean of cluster and the centroid of cluster. Then the developed updated mean of cluster is given by Eq.36:

$$\mathbf{c}'_j(t+1) = \mathbf{c}'_j(t) + \eta_2(t)\left[\frac{1}{n_j} \sum_{\mathbf{x}_i(t) \in \mathbf{c}'_j} \mathbf{x}_i(t) - \mathbf{c}'_j(t)\right] \quad (36)$$

where $\mathbf{c}'_j(t)$ is calculated using Eq.29 and denotes:

$$\Delta_2 = \eta_2(t)\left[\frac{1}{n_j} \sum_{\mathbf{x}_i(t) \in \mathbf{c}'_j} \mathbf{x}_i(t) - \mathbf{c}'_j(t)\right] \quad (37)$$

where $0 < \eta_2(t) < 1$, $\Delta_2$ in Eq.37 is the distortion between the mean of cluster and the centroid of cluster from Eq.36. With the same form in Eq.35 and Eq.37 and assuming that the updated means are the same in time of $t$, the distortion is decided by the clustering rate $\eta_1$ and $\eta_2$ respectively. Since $\eta_1 > \eta_2$ as defined, the result $|\Delta_2| < |\Delta_1|$ can be concluded. Therefore, the distortion of the mean by the proposed clustering is always less than the distortion by the original batch clustering. As

shown in Fig. 1, 'Δ' represents the ideal mean of one spectral cluster *j*, where $j = 1, 2,\ldots,k$. The initial mean of the cluster '*' is given by the estimation process in the first part. The solid point '•' represents the updated mean of the cluster by the original c-means batch clustering. The circle point '°' represents the updated mean of the cluster by the proposed clustering process. There are three cases to be considered as shown graphically. The first case considers the location of the updated mean in the negative side with respect to the ideal and initial or pre-updated mean. The second case considers the location of the updated mean in the positive side of the initial or pre-updated and the ideal mean. These two cases for the new updating rule clearly draw the mean of the cluster closer to the ideal mean of the cluster by introducing the gain scalar which is less than the unit. The third case considers the location of the updated mean between the ideal and the initial or the pre-updated mean. In third case, the updated mean conforms to the original convergence process. The distortion is adjusted with the attenuated clustering gain scalar. It is clear that in case three the updated parameter follows the one updated with original clustering rule. Therefore, by combining all cases, the updated mean by the proposed clustering process is always driven closer to the ideal mean of the cluster thus to reduce the drifting issue.

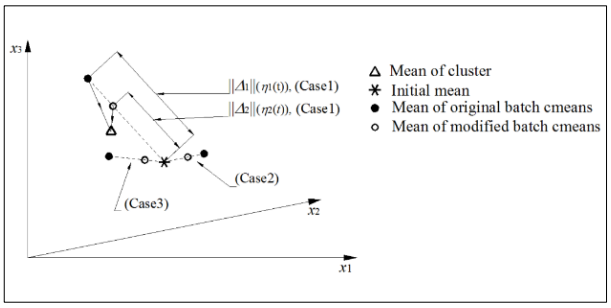


Fig. 1. *Difference of updated mean between the modified batch clustering and original c-means batch clustering process*.

Considering all possible cases, the clustering gain $\eta_2(t)$ is selected to be decreasing monotonically to the infinitesimal by starting with the initial clustering gain $\eta(t) = \eta_0 \exp(-t/T)$, where $T$ is the bound of iteration and the initial clustering gain $\eta_0 = (0,1]$. The empirical thumb rule can be used for $T$ selection where the bound of iteration is a multiple of the size of initial number of data clusters. Regardless of the variants of SGD such as the momentum or learning gain, the decreasing gain scalar guarantees that the updated mean converge to the ideal mean. However, the main contributor for the speed-up is due to the scenario of case 1 and case 2 during the clustering process since the distortion always drives the parameter closer to ideal centroid of the hypersphere. The complexity of the performance of each iteration has no more change than about O(*nkm*) where *n* is the number of instance, *k* is the number of parameters and *m* is dimension of vector. The factor for the speed-up of convergence is mainly the modification by directing the parameter with respect to the updated mean to the minimum of search method with gradient descent to reduce the drifting phenomena. The process of the gradient descent batch clustering in Table 2 is iterated until the mean square error of the updated mean reaches finite partial minimum in fast manner.

Table 2. *Algorithm 2 for gradient descent back clustering process for refining the means of the clusters*.

Input 1: Image instance matrix.

Input 2: Estimated initial mean of the clusters matrix

Input 3: Error criterion for termination.

Output: Converged mean of clusters in matrix for classification.

1. Initialize $t = 1$, $T$ is positive integer for adjusting clustering gain.

2. while $MSE > Error\ criteria$

3. Adjust scalar $\eta = \eta_0 \times \exp(-t/T)$

4. Find number $n_j$ of instance in cluster *j* by $\min \|\mathbf{x}_i - \mathbf{c}_j\|, \quad i = 1,...,n, \quad j = 1,...,k$

5. $\mathbf{c}'_j = \frac{1}{n_j} \sum_{\mathbf{x}_i \in \mathbf{c}_j} \mathbf{x}_i, \quad i = 1,...,n_j, \quad j = 1,...,k$ %

Update mean based on current mean.

6. $\mathbf{c}_j(t+1) = \mathbf{c}_j(t) + \eta\left[\mathbf{c}'_j(t) - \mathbf{c}_j(t)\right]$

7. *MSE* calculation

8. $t = t + 1$

9. End

## VALIDATION OF GRADIENT DESCENT BATCH CLUSTERING

The proposed method is validated with two stages of experimental study. In the first step, some benchmark images available from Columbia multispectral image database Asuni *et al*. (2014) are selected and used for the comparison between the proposed method and the other original algorithms. Further the citrus fruit images captured under the natural lighting conditions are used in the second stage of the study. Three types of citrus color images are used: normal color image without any filters (termed VIS images), neutral filtered color images (termed NEUT images), and the linear polarizer filtered images (termed POLA images).

The evaluation study is conducted with Matlab. Three variants of C-means clustering methods, namely the single data based gradient descent clustering (named as method1), the original batch clustering (named as method2) and the proposed gradient descent batch clustering (named as method3) are used in the validation study. After the initial value of clusters centroids are identified, they are refined using C-means variant clustering methods. After the clustering process converges to the finite partial minimum, the refined centroids are used to label the clusters in the image. The number of clusters estimated varied based on how fluently the background features are. As for the multispectral color image database, the standard color bars or the one with few blobs have cluster number estimated with certainty as shown in table 2. As for the natural citrus color image, the parameters are varied due to the unstructured background color. Hence only those salient color clusters are used to check the dissimilarity with the manually created ground truth references using *F* measure. The expectation of the accuracy measure for the variants of clustering is close based on the same pack of initial parameters. However the efficiency of the clustering process by the variants should be differed from each other.

Table 3. *Processing time (in second) for some color images.*

| Method | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 | a9 | a10 | a11 | a12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method1 | 2.33 | 2.71 | 8.07 | 5.75 | 10.96 | 4.14 | 41.88 | 5.30 | 6.05 | 8.28 | 20.56 | 5.97 |
| Method2 | 0.23 | 3.47 | 6.25 | 3.63 | 3.22 | 0.80 | 8.37 | 2.09 | 14.70 | 2.79 | 2.72 | 9.86 |
| Method3 | 0.22 | 0.79 | 0.87 | 0.98 | 1.05 | 0.37 | 2.34 | 1.18 | 1.72 | 1.09 | 0.74 | 1.21 |
| No. of clusters | 8 | 9 | 12 | 17 | 12 | 7 | 21 | 7 | 28 | 1 | 19 | 16 |

a1: Colour standard bars; a2: Balloons; a3: Beads; a4: Clay; a5: Feathers; a6: Colour Bars; a7: Glass tiles; a8: Jelly Beans; a9: Paints; a10: Pompoms; a11: Sponges; a12: Superballs;

Table 4. *Processing time (in second) for citrus color image with three Cmens methods.*

| Cmeans variant methods | VIS | NEUT | POLA |
|---|---|---|---|
| Method1 | 116.41 | 114.57 | 151.82 |
| Method2 | 77.85 | 70.96 | 78.73 |
| Method3 | 15.81 | 15.31 | 14.85 |
| No. of clusters estimated | 8 | 19 | 21 |
| STD of parameters | 7 | 7 | 9 |

Method1: single data based gradient descent clustering process
Method2: original Cmeans batch clustering process
Method3: gradient descent batch clustering process

As a result the efficiency of the clustering procedure is given in Table 3 and Table 4 for two streams of color image data. As shown in the table the modified gradient descent batch clustering is much faster than other two clustering algorithms. The speed is significant even on each applied citrus fruit image data with variance based on the size of initial parameters estimated from the estimation method in the first part. Since the objective of the algorithm is to converge the parameters into the ideal ones, due to the dimension of the parameter vector for illustration, the distortion between the mean of cluster and the parameter of centroid is measured in MSE as shown graphically in Fig. 4 with one example. In matrix the parameters are updated altogether. Hence when MSE of all parameters converge to finite minimum, it is agreed that the parameters converge to the ideal centroids of clusters. As shown in the figure both batch clustering algorithms (as shown in solid line and dash line) converge to finite minimum rapidly without much oscillation. Since the decimal level is largely different in graphic with other two methods, hence MSE by SGD batch clustering algorithm is re-generated separately in Fig. 5. The modified clustering algorithm converges more rapidly in decimal level by comparing with the original batch clustering algorithm. As for the simple example data based gradient descent cluster, the random selection of sample data from the whole image is used for the competitive rule. However it is clearly shown that the simple example data based gradient descent clustering still oscillates to further iterations. The results show that the change of the distortion of the means by the gradient descent batch clustering can speed up the clustering convergence with the estimated initial parameters.

The clustering accuracy measure for two types of color image database are given in Fig. 2 and Fig. 3 graphically. The *F* measure can be read in Table 5 and Table 6. As expected the benchmark color images with standard color bars or less noise blobs have the same or very close *F* measure. Those with feature-fluent background noise are all acceptable even with some images having higher *F* measure comparably. The natural citrus

fruit images also have similar *F* measure for each type of color image averagely. The drop down of the *F* measure for the attenuated color image is due to the loss of the citrus fruit area by the attenuation function. Due to the non-Euclidean posture of the fruit body, the color of real time citrus fruit is non-homogeneous. Hence some area of fruit part is excluded into neighboring cluster thus to drop *F* measure in result. However the *F* measure for each type of citrus color image are very close. On top of the mathematical justification, the empirical study further validates that the change of the distortion between the mean of cluster and the parameter using SGD direction to the minimum can possibly drive the mean of cluster to the partial optimum more efficiently. Therefore the overall performance proves the speed-up of the convergence to the partial optimum by the SGD batch clustering based on the initialization of the parameters.

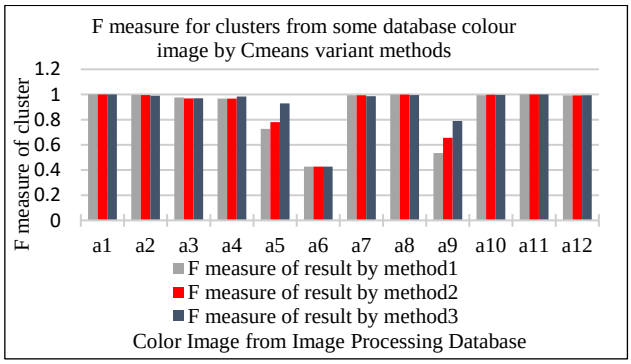


Fig. 2. *F measure of clustering results for some color image database by three C-means methods.*

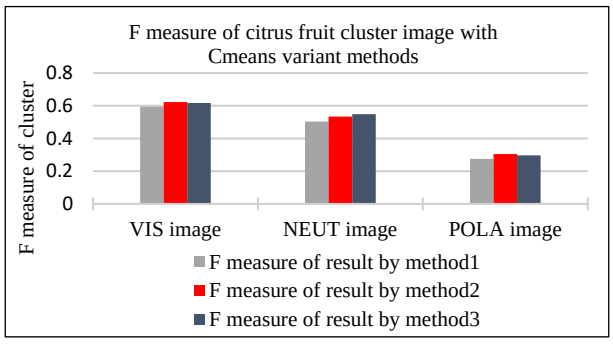


Fig. 3. *F measure of clustering results for citrus color image by three C-means methods.*

Table 5. *F measure for standard color image database (Alpha:0.7).*

| Image / Method | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 | a9 | a10 | a11 | a12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method1 | 1.0 | 0.9965 | 0.9765 | 0.9670 | 0.7266 | 0.4276 | 0.9936 | 0.9988 | 0.5349 | 0.9934 | 1.0000 | 0.9926 |
| Method2 | 1.0 | 0.9950 | 0.9692 | 0.9668 | 0.7801 | 0.4276 | 0.9940 | 0.9989 | 0.6554 | 0.9977 | 1.0000 | 0.9932 |
| Method3 | 1.0 | 0.9906 | 0.9693 | 0.9844 | 0.9296 | 0.4276 | 0.9859 | 0.9955 | 0.7898 | 0.9962 | 1.0000 | 0.9942 |

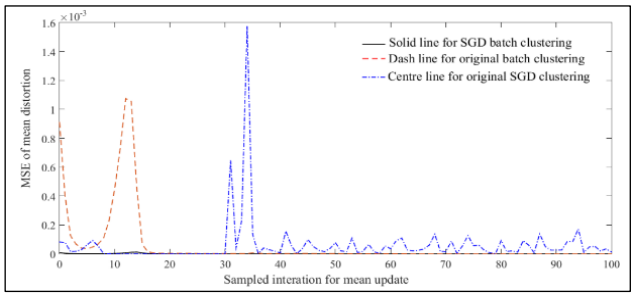


Fig. 4. *MSE of average mean distortion from three C-means clustering variant algorithms.*

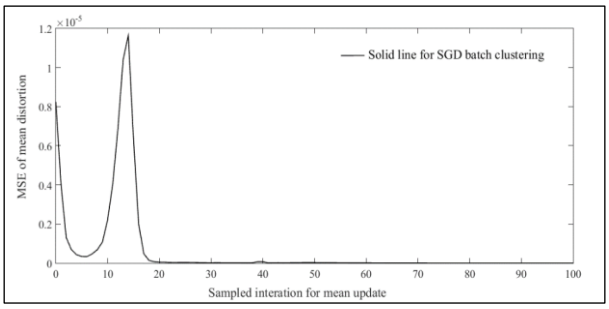


Fig. 5. *MSE of average mean distortion from SGD batch clustering algorithm.*

## CONCLUSIONS

In this paper an unsupervised clustering method has been proposed for the image data classification application. The clustering method consists of both the initialization of parameters and the gradient descent batch clustering to speed up the refining clustering process. The parameters are estimated based on a mathematical connection between PCA and C-means clustering membership. The nonlinear centroids parameters are estimated with a hierarchical PCA solution along with the global threshold iteratively. Then the initial parameters are refined by the proposed gradient descent batch clustering process. The modified clustering algorithm applies a gradient descent on the objective to direct the parameter to the partial optimum in search line. Therefore the drifting of the original stochastic batch clustering is reduced based on both the mathematical justification and the validation study. The validation results with statistical F measure proves a significant improvement of the efficiency with tradeoff to the accuracy by the modified gradient descent batch clustering.

## ACKNOWLEDGMENT

This work was supported by the Incheon National University (International Cooperative) Research Grant in 2019.

## REFERENCES

Asuni N. and Giachetti A, Testimage: a large-scale archive for testing visual devices and basic image processing algorithms. 2014, STAG: Smart Tools & Apps for Graphics.

BandyoPadhyay S (2001), 'Clustering using simulated annealing with probabilistic redistribution', Internation J Pattern Recogn Artif Intell, vol.15, no.2, pp.269-85.

Chen B, et al. (2005), 'Novel Hybrid Hierarchical-K-means Clustering Method (H-K-means) for Microarray Analysis', Computational Systems Bioinformatics Conference 2005. Stanford University, IEEE, pp.105-8.

Delport V (1996), 'Codebook design in vector quantisation using a hybrid system of parallel simulated annealing and evolutionary selection', Electron Lett, vol.32, no.13, pp.1158-60.

Du K-L. and Swamy MNS (2006), 'Neural Networks in a Softcomputing Framework', London, Springer-verlag London Limited.

Duda RO, Hart PE, and Stork DG (2000), 'Pattern Classification', 2nd, New York, Wiley-Interscience.

Ester, M., et al. (1996). 'A density-based algorithm for discovering clusters in large spatial databased with noise', KDD-96 Proceedings, 226-31.

Guha S, Rastogi R, and Shim K (2000), 'Rock: A robust clustering algorithm for categorical attributes', Information Systems, vol.25, no.5, pp.345-66.

Guha, S., R. Rastogi, and K. Shim (1998). 'CURE: An efficient clustering algorithm for large databases', Proc. ACM SIGMOD Int. Conf. Management of Data, 73-84.

Gray NH, Anderson JD, Devine JD, Kwasnik JM (1976). Topological properties of random crack networks. Math Geol 8:617–26.

Haykin S (1999), 'Neural Network A Comprehensive Foundation', 2nd, New Jersey, Tom Robbins.

Ioffe S and Szegedy C (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, **arXiv:1502.03167**, 2008.

Jimenez AR, Ceres R, and Pons JL (2000). 'A Survey of Computer Vision Methods for Locating Fruit on Trees', Transactions of the ASAE, vol.43, no.6, pp.1911-20.

Karypis G, Han E-HS, and Kumar V (1999), 'Chameleon: hierarchical clustering using dynamic modeling', Computer, vol.32, no.8, pp.68-75, Issn:0018-9162, Doi:10.1109/ 2.781637.

Khan F, 'An initial seed selection algorithm for k-means clustering of georeferenced data to improve replicability of cluster assignments for mapping application', Applied Soft Computing, vol.12, 2012, pp.3698-700.

Kohonen T (2001), 'Self-Organizing Maps', 3rd, Berlin, Springer.

Kohonen T (1990). 'The Self-Organizing Map', Proceedings of the IEEE, vol.78, 1464-80.

Krishna K, and Murty MN (1999), 'Genetic K-Means Algorithm', IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, vol.29, no.3, pp.433-39.

Lai JZC, and Liaw Y-C (2008), 'Improvement of the k-means clustering filtering algorithm', Patt Recogn, vol.41, pp.3677-81.

Li P, Lee S-H, and Hsu H-Y (2012), 'Fusion on Citrus Image Data from Cold Mirror Acquisition System', Int. J Comput Vis Image Process, vol.2, no.4, pp.12-26, Issn:2155-6997, Doi:10.4018/ijcvip.2012100102.

Li P, Lee S-H, and Hsu H-Y (2011). 'Study on citrus fruit image data separability by segmentation methods', 2011 International Conference on Power Electronics and Engineering Application. Shenzhen, China, Procedia Engineering Elsevier, vol.23, 408-16, Isbn:1877-7058.

Linde Y, Buzo A, and Gray RM (1980), 'An Algorithm for Vector Quantizer Design', IEEE Trans Commun, vol.COM-28, no.1, pp.84-95.

MACQUEEN J (1967). 'Some methods for classification and analysis of multivariate observations', 5th Berkeley Symp on Math Statistics and Probability. University of California Press, Berkeley, 281-97.

Patane G, and Russo M (2001), 'The enhanced LBG algorithm', Neural Networks, vol.14, pp.1219-37.

Qian Y, et al. (2016), 'Space Structure and Clustering of Categorical Data', IEEE Trans Neural Netw Learn Syst, vol.27, no.10, pp.2047-59, Doi: 10.1109/TNNLS.2015.2451151.

Sujatha S, and Sona AS, 'New fast K-means clustering algorithm using modified centroid selection method', Int J Eng Res Technol (IJERT), vol.2, no.2, 2013, pp.1-9, Issn:2278-0181.

Xiang T, and Gong S, 'Spectral clustering with eigenvector selection', Patt Recogn, vol.41, 2008, pp.1012-29.

Xu R, and Wunsch D (2005), 'Survey of Clustering Algorithms', IEEE Trans Neural Netw, vol.16, no.3, pp.645-78.

Zaremba W, and Sutskever I, 'Learning to Execute', Internationa conference on Learning Representations, 2015.

Zhang T, Ramakrishnan R, and Livny M (1996). 'BIRCH: An Efficient Data Clustering Method for Very Large Databases', ACM SIGMOD Conf. Management of Data, 103–14.