Commonality Preserving Multiple Instance Clustering Based On Diverse Density

Takayuki Fukui and Toshikazu Wada

Graduate School of Systems Engineering,
Wakayama University, Japan

Abstract. Image-set clustering is a problem decomposing a given image set into disjoint subsets satisfying specified criteria. For single vector image representations, proximity or similarity criterion is widely applied, i.e., proximal or similar images form a cluster. Recent trend of the image description, however, is the local feature based, i.e., an image is described by multiple local features, e.g., SIFT, SURF, and so on. In this description, which criterion should be employed for the clustering? As an answer to this question, this paper presents an image-set clustering method based on commonality, that is, images preserving strong commonality (coherent local features) form a cluster. In this criterion, image variations that do not affect common features are harmless. In the case of face images, hair-style changes and partial occlusions by glasses may not affect the cluster formation. We defined four commonality measures based on Diverse Density, that are used in agglomerative clustering. Through comparative experiments, we confirmed that two of our methods perform better than other methods examined in the experiments.

1 Introduction

Image-set clustering is a problem dividing a given image set into disjoint image subsets (clusters). Images belonging to a cluster should satisfy some criteria, e.g. mutual similarity or proximity. The image-set clustering can be utilized for many applications. For example, the clustering can be used as a chunking process of training images for accelerating the learning process of large-scale image classification systems. Also, the cluster information helps image annotation, labeling, and arranging the pictures to create photo albums.

For single vector image descriptions, similarity or proximity criterion is widely applied, i.e., mutually similar images (proximal image vectors) form a cluster. However, recent trend of the image description is the local feature based, i.e., an image is described by multiple local features, e.g. SIFT[1], SURF[2], and so on. Such image descriptions have some advantages over single-vector description, i.e., robustness against occlusions and geometric transformations. This description principle can be generalized from local features to any other features. That is, an image can be described by arbitrary number of vectors depending on the amount of intrinsic information in the image. In this description, which criterion should be used for the clustering?

Of course, Bag of Features (BoF)[3] is a useful description that enables us to use many algorithms developed for the single-vector image description. If we employ BoF description, we can keep relying on similarity or proximity based clustering algorithms. BoF vector description, however, requires large scale feature-vector clustering to create a code book (visual words), which accumulates the occurrence of local features. This clustering consumes significant computational time and memory. Also, since the performance depends on the code book size, we have to run the vector clustering many times by changing the codebook size. Because of these reasons, we stick on the question above, i.e., which criterion should be used for image-set clustering for multiple-vector image description.

This paper responds to the question that *commonality* of the feature vectors can be a criterion of image-set clustering. The term “commonality” in this paper means the number and/or strength of commonly existing features across the image set. Basically, similarity or dissimilarity is defined between two images, but commonality measure is naturally defined on an image set. From this viewpoint, it is clear that commonality and similarity are essentially different criteria.

Based on this idea, we propose an image-set clustering method, i.e., images preserving strong commonality (coherent feature vectors) form a cluster. This clustering method has the robustness against common feature preserving image variations. In the case of face images, hair-style changes and partial occlusions by glasses rarely affect the clustering result.

We first define four commonality measures based on Diverse Density (DD)[4, 5]. These measures are used in agglomerative (bottom-up) clustering that iteratively merges the image-set pair having the maximum commonality measure to create hierarchical image-set clusters. Thus, we get four clustering methods. Through comparative experiments conducted on “the ORL database of face” consisting of 400 face images and a part of “Nister’s image dataset”, we confirmed that two of our methods are almost competitive and perform better than other methods including k-means++[6] and Ward’s clustering[7] applied to BoF vectors and Hausdorff clustering applied to multiple-vector image descriptions.

2 Related Works

Clustering methods can be classified into two types “partitional” and “hierarchical” methods[8]. Partitional clustering divides given dataset into several subsets without checking all possible subset systems. On the contrary, hierarchical clustering builds possible cluster hierarchy, which is represented by tree-shaped data structure known as *dendrogram*. Of course the partition algorithm is more efficient than hierarchical one, because all possible subsets are not examined. But, this drawback of hierarchical clustering can be relaxed by introducing stop conditions.

K-means clustering[9, 10]and k-medoid clustering[11] are the popular partition clustering methods under the assumption that the number of clusters is known. In contrast to the k-means clustering designed for vector data, k-medoids chooses data points as cluster centers (medoids or exemplars) so as to work in any metric space. For vector data, k-means clustering is one of the most popular method, but it depends on the initial locations of the cluster centers. For solving this problem, k-means++[6] (k-means clustering with cluster center initialization) has been proposed.

Hierarchical clustering can further be classified into agglomerative (bottom-up) and divisive (top-down) methods [12]. Both methods require *linkage metric*: agglomerative clustering iteratively merges cluster pairs having minimum linkage metric, and divisive clustering extracts the subcluster pairs having maximum linkage metric by separating a cluster. The computational cost of divisive clustering is more expensive than agglomerative clustering, because of the bigger number of separation cases.

Based on the discussion above, we can notice that agglomerative clustering is one of the simplest clustering method. It only requires linkage metric, and the computational cost is not so big. Because of this simplicity, we employ this algorithm and focus on the linkage metric design.

Most linkage metric is obtained by extending the between-image distance to between-image-set distance. This framework is represented by Lance-Williams dissimilarity updating formula[13]. This formula is defined as

$$d(i \cup j, k) = \alpha_i d(i,k) + \alpha_j d(j,k) + \beta d(i,j) + \gamma \left| d(i,k) - d(j,k) \right|, \tag{1}$$

where i, j and k represent image clusters, α_i, α_j, β, and γ are the parameters that define agglomerative criterion, and $|\cdot|$ represents the absolute value. Each image clusters have centers $\boldsymbol{g}_i$, $\boldsymbol{g}_j$ and $\boldsymbol{g}_k$, and the between-class dissimilarity is defined based on the distance between cluster centers. Just by assigning these parameters, dissimilarity definition, and cluster center update formula, wide varieties of agglomerative clustering algorithms can be described [14]. For example, Ward's clustering method[7], which merges the cluster pair so as to minimize the variance of the merged cluster, can be described by the following settings.

$$\alpha_i = \frac{|i| + |k|}{|i| + |j| + |k|},\ \beta = -\frac{|k|}{|i| + |j| + |k|},\ \gamma = 0, \tag{2}$$

$$\boldsymbol{g}_{i \cup j} = \frac{|i|\boldsymbol{g}_i + |j|\boldsymbol{g}_j}{|i| + |j|},\ d(i,j) = \frac{|i||j|}{|i| + |j|} \left\| \boldsymbol{g}_i - \boldsymbol{g}_j \right\|^2, \tag{3}$$

where $|\cdot|$ and $\|\cdot\|$ represents the number of elements in the cluster and Euclidian norm, respectively.

The other approach is to employ between-set distance in agglomerative clustering. The most popular one is Hausdorff distance. This measure is the greatest of all the distances from a data point in one set to the closest data point in the other set. This metric is often used in the template matching scenario, because the metric is inverse proportional to the resemblance of two dataset distributions[15]. For the image clustering, however, Hausdorff distance is not suitable because of its excessive sensitivity to the outliers. For example, if a feature vector leaps into one set at the furthest point from the other set, the Hausdorff distance between them changes. That is, only one vector may drastically change the distance between two sets.

Same as the linkage metric, similarity measure can be defined and used in agglomerative clustering. This approach has wider variations than metric approach. This is because the similarity measure has lesser axioms than metrics and bigger degree of freedom. Furthermore, just by providing similarity values between any pairs instead of defining explicit form of similarity function, we can perform clustering like affinity propagation[16].

A question arose here, how do we define the linkage metric for multi-vector representation of images? Of course, when we employ Bag of Features (BoF)[3] image description, we can keep using standard clustering methods, such as k-means++ or Ward's method. The other possibility is the Hausdorff clustering between two sets of feature vectors. The method, however, is not suitable for linkage metric, because one image may contain wide variety of vectors and the Hausdorff distance is very sensitive to the outliers, as mentioned above.

As discussed above, no effective clustering method for multiple-vector image description has been proposed so far.

In the following sections, we propose commonality preserving clustering in Section 3, comparative experiments of image-set clustering conducted over "the ORL database of face" and a part of "Nister's image dataset" are shown and discussed in Section 4, and we conclude the discussion in Section 5.

3 Commonality Preserving Image-Set Clustering

In this section, we define a commonality measure between two image sets for agglomerative clustering. This commonality measure is defined based on Diverse Density (DD) $DD(\boldsymbol{x})$, which represents how the feature $\boldsymbol{x}$ commonly appears in positive bags (images) and never appears in negative bags. By integrating the value $DD(\boldsymbol{x})$ in the whole feature space, we can define the commonality measure. In some cases, it may be necessary to find the maxima of $DD(\boldsymbol{x})$ for common feature extraction from positive bags. For such computation, EM-DD has been proposed. This is an accelerated hill-climbing method of $DD(\boldsymbol{x})$ in the feature space. In this section, we shortly introduce both DD and EM-DD, and define four commonality measures based on them.

3.1 Diverse Density

In the field of MIL[4, 5], common feature extraction has been regarded as one of the essential problems, which can be formalized as the local maxima search problem of DD in the feature space. Compared with other sophisticated MIL methods, DD is very sensitive to the incorrect labeling. That is, if a positive feature leaps into a negative bag (image) or a feature is removed from a positive bag (image), the value of DD may change drastically. This sensitivity is unwanted property for many MIL applications that use manually labeled data. But for commonality preserving clustering, this sensitivity is welcome, because the task is not extracting features from incorrectly labeled data, but gathering images preserving strong commonality.

Based on the following definitions and terminology, we define the DD.

Bag $\mathcal{B}$: A set of instances. This corresponds to an image in our problem.

Label $+, -$: We assign positive labels to those bags where we want to found out the commonality. Also, negative labels are assigned to those bags where the commonality never be expected. These are denoted by $\mathcal{B}_i^+, (i = 1, \ldots, m)$ and $\mathcal{B}_i^-, (i = 1, \ldots, n)$, respectively.

Instance $\boldsymbol{B}_{ij}^+, \boldsymbol{B}_{ij}^-$: An element belonging to a bag. This corresponds to a local feature vector. Positive and negative instances are denoted by $\boldsymbol{B}_{ij}^+ \in \mathcal{B}_i^+$ and $\boldsymbol{B}_{ij}^- \in \mathcal{B}_i^-$, respectively.

First, the following function represents a potential generated by an instance $\boldsymbol{B}_{ij}$ at a point $\boldsymbol{x}$ in feature space.

$$P(\boldsymbol{x}|\boldsymbol{B}_{ij}) \equiv \exp\left(-\frac{\|\boldsymbol{B}_{ij} - \boldsymbol{x}\|^2}{\sigma^2}\right). \tag{4}$$

The maximum and the minimum values of this potential are 1 and 0, respectively.

The following function represents the integrated potential $P(\boldsymbol{x}|\mathcal{B}_i^+)$ generated by instances in a positive bag $\mathcal{B}_i^+$.

$$P(\boldsymbol{x}|\mathcal{B}_i^+) \equiv 1 - \prod_{\boldsymbol{B}_{ij}^+ \in \mathcal{B}_i^+} \left(1 - P\left(\boldsymbol{x}|\boldsymbol{B}_{ij}^+\right)\right). \tag{5}$$

Subtraction of an individual potential from 1 can be regarded as the similar meaning to negation and the product can be regarded as logical AND. Under this interpretation, Equation (5) can be regarded as integration by logical OR of the individual potentials in the positive bag by applying De Morgan's laws.

For negative bag, integrated potential from a negative bag $\mathcal{B}_i^-$ can be defined as follows.

$$P(\boldsymbol{x}|\mathcal{B}_i^-) \equiv \prod_{\boldsymbol{B}_{ij}^- \in \mathcal{B}_i^-} \left(1 - P\left(\boldsymbol{x}|\boldsymbol{B}_{ij}^-\right)\right). \tag{6}$$

Same as the interpretation of Equation (5), this integration can be regarded as logical NOR.

The potentials generated by positive and negative bags are further integrated by their product to obtain Diverse Density $DD(\boldsymbol{x})$.

$$DD(\boldsymbol{x}) \equiv \prod_i^m P\left(\boldsymbol{x}|\mathcal{B}_i^+\right) \prod_i^n P\left(\boldsymbol{x}|\mathcal{B}_i^-\right). \tag{7}$$

At a local maximum of $DD(\boldsymbol{x})$ having enough value in the feature space, the point $\boldsymbol{x}$ can be regarded as a common local feature among the positive bags and does not contain similar features in all negative bags.

By integrating the $DD(\boldsymbol{x})$ value in whole feature space, we can measure the commonality of the local features in a positive image set.

3.2 EM-DD

$DD(\boldsymbol{x})$ can be regarded as a commonality measure at a point $\boldsymbol{x}$ in the feature space. We sometimes need the points that locally maximize $DD(\boldsymbol{x})$ for determining the features specifying the given cluster (positive and negative image sets). EM-DD[17] estimates a local maximum of DD in the feature space by hill climbing iterations, in which an accelerated approximation of DD like EM-algorithm[18] is employed. EM-DD algorithm iteratively finds local maxima in the feature space starting from all positive instances. The DD is defined in Equation (7), but the computation using all positive and negative instances is cumbersome. For avoiding this, EM-DD approximates DD value by using proximal instances, each of which is the 1-nearest instance picked up from a positive or negative bag. Since this selection process is similar with expectation process, and the hill climbing can be regarded as maximization process, this algorithm is called EM-DD.

3.3 Commonality Between Two Clusters

In this section, we discuss how to define the commonality measure between two image sets. The measure should represent how many and/or strong common local features are preserved after merging two image sets. According to this principle, the measure $\mathcal{C}$ can be easily defined as follows.

$$\mathcal{C}_{\mathbb{N}}(\mathbb{A} \cup \mathbb{B}) \equiv \int_{\boldsymbol{x} \in \mathcal{F}} DD(\boldsymbol{x}) d\boldsymbol{x}, \tag{8}$$

where $\mathbb{A}$ and $\mathbb{B}$ represent positive image sets, $\mathbb{N}$ negative image set, and $\mathcal{F}$ feature space. This commonality $\mathcal{C}_{\mathbb{N}}(\mathbb{A} \cup \mathbb{B})$ is obtained by integrating $DD(\boldsymbol{x})$ over feature space $\mathcal{F}$. However, this computation is practically impossible, because the feature space $\mathcal{F}$ is infinitely vast.

For avoiding this endless computation, the computation of $DD(\boldsymbol{x})$ can be restricted on sampling points. In this case, one positive image set is assigned as the positive image set and the other is used to define the sampling point in the feature space. So as to produce a commonality measure that is independent of number of sampling points, we compute the sample mean of $DD(\boldsymbol{x})$. By assigning the image set $\mathbb{A}$ to produce sampling point set $\mathcal{S}_{\mathbb{A}}$ in feature space and $\mathbb{B}$ as positive image set, we get

$$\mathcal{C}_{\mathbb{N}}^{\mathcal{S}_{\mathbb{A}}}(\mathbb{B}) \equiv \frac{1}{|\mathcal{S}_{\mathbb{A}}|} \sum_{\boldsymbol{x} \in \mathcal{S}_{\mathbb{A}}} DD(\boldsymbol{x}), \tag{9}$$

where $\mathcal{S}_{\mathbb{A}}$ represents the feature set extracted from all images in the image set $\mathbb{A}$. This computation is possible, but the resulted value has asymmetric property depending on the assignment as shown in Equations (10).

$$\mathcal{C}_{\mathbb{N}}^{\mathcal{S}_{\mathbb{A}}}(\mathbb{B}) \neq \mathcal{C}_{\mathbb{N}}^{\mathcal{S}_{\mathbb{B}}}(\mathbb{A}). \tag{10}$$

Algorithm 1 Agglomerative clustering

```
Initialize: Create singleton clusters, each of which consists of a single image.
Step1: Merge the cluster pair having maximum commonality measure among all cluster pairs.
Step2: If the number of cluster is greater than one, go to Step 1.
End:
```

For guaranteeing the symmetric property, we first define a commonality measure by taking arithmetic mean as

$$\mathcal{C}^1_{\mathbb{N}}(\mathbb{A},\mathbb{B}) \equiv \frac{1}{2}(\mathcal{C}^{\mathcal{S}_{\mathbb{B}}}_{\mathbb{N}}(\mathbb{A}) + \mathcal{C}^{\mathcal{S}_{\mathbb{A}}}_{\mathbb{N}}(\mathbb{B})). \tag{11}$$

We can also define other commonality measure by taking geometric mean as

$$\mathcal{C}^2_{\mathbb{N}}(\mathbb{A},\mathbb{B}) \equiv \sqrt{\mathcal{C}^{\mathcal{S}_{\mathbb{B}}}_{\mathbb{N}}(\mathbb{A}) * \mathcal{C}^{\mathcal{S}_{\mathbb{A}}}_{\mathbb{N}}(\mathbb{B})}. \tag{12}$$

The above definitions require excessive sampling points in the feature space for big image data sets. For avoiding this problem, we can reduce the number of sampling points by applying EM-DD to image set $\mathbb{A}$ to produce reduced sampling point set $\mathcal{M}_{\mathbb{A}}$. By using this as a point set, we can define other commonality measure as

$$\mathcal{C}^{\mathcal{M}_{\mathbb{A}}}_{\mathbb{N}}(\mathbb{B}) \equiv \frac{1}{|\mathcal{M}_{\mathbb{A}}|} \sum_{\boldsymbol{x}\in\mathcal{M}_{\mathbb{A}}} DD(\boldsymbol{x}). \tag{13}$$

Same as the discussion above, we propose the following commonality measures:

$$\mathcal{C}^3_{\mathbb{N}}(\mathbb{A},\mathbb{B}) \equiv \frac{1}{2}(\mathcal{C}^{\mathcal{M}_{\mathbb{B}}}_{\mathbb{N}}(\mathbb{A}) + \mathcal{C}^{\mathcal{M}_{\mathbb{A}}}_{\mathbb{N}}(\mathbb{B})), \tag{14}$$

$$\mathcal{C}^4_{\mathbb{N}}(\mathbb{A},\mathbb{B}) \equiv \sqrt{\mathcal{C}^{\mathcal{M}_{\mathbb{B}}}_{\mathbb{N}}(\mathbb{A}) * \mathcal{C}^{\mathcal{M}_{\mathbb{A}}}_{\mathbb{N}}(\mathbb{B})}. \tag{15}$$

As discussed above, we propose four commonality measures, $\mathcal{C}^1_{\mathbb{N}}(\mathbb{A},\mathbb{B})$, $\mathcal{C}^2_{\mathbb{N}}(\mathbb{A},\mathbb{B})$, $\mathcal{C}^3_{\mathbb{N}}(\mathbb{A},\mathbb{B})$, $\mathcal{C}^4_{\mathbb{N}}(\mathbb{A},\mathbb{B})$ in this paper. These commonality measures are used in the Algorithm1.

4 Experiments

We conducted comparative experiments on image-set clustering. We first introduce the normalized mutual information (NMI)[19] for evaluating the accuracy of the clustering results. Next, we describe the experimental settings and parameter tuning of different methods (Only the resulted parameters are shown in this paper because of the page limitation). Finally, we compared NMI scores by changing number of clusters, and examine images in resulted clusters.

4.1 Evaluation Criterion

We employ NMI for measuring the clustering accuracy. Let $\mathbb{X}$ be the set of clusters obtained by a clustering algorithm and $\mathbb{Y}$ obtained from the ground truth. Then, the NMI measure between $\mathbb{X}$ and $\mathbb{Y}$ is defined as

$$NMI(\mathbb{X}, \mathbb{Y}) = \frac{I(\mathbb{X}; \mathbb{Y})}{(H(\mathbb{X}) + H(\mathbb{Y}))/2}, \tag{16}$$

where

$$I(\mathbb{X}; \mathbb{Y}) = H(\mathbb{X}, \mathbb{Y}) - H(\mathbb{X}|\mathbb{Y}) - H(\mathbb{Y}|\mathbb{X}), \tag{17}$$

is the mutual information between $\mathbb{X}$ and $\mathbb{Y}$, $H(\mathbb{X}, \mathbb{Y})$ and $H(\mathbb{X}|\mathbb{Y})$ are the joint and conditional entropies of $\mathbb{X}$ and $\mathbb{Y}$, respectively. When $\mathbb{X}$ and $\mathbb{Y}$ are the same set of clusters, $NMI(\mathbb{X}, \mathbb{Y})$ is 1. Conversely, When $NMI(\mathbb{X}, \mathbb{Y})$ is nearly 0, $\mathbb{X}$ and $\mathbb{Y}$ are different set of clusters. By using $NMI(\mathbb{X}, \mathbb{Y})$, we evaluate the clustering accuracy.

4.2 Experimental settings

The experimental settings are shown below.

Methods to be Compared

In the experiments, we compare the following clustering methods. K-means++ and Ward's clustering are applied to BoF vectors.

Agglomerative Clustering with $\mathcal{C}^1_{\mathbb{N}}(\mathbb{A}, \mathbb{B})$: $AM\_ALL(\sigma)$, where σ represents the parameter in Equation(4).

Agglomerative Clustering with $\mathcal{C}^2_{\mathbb{N}}(\mathbb{A}, \mathbb{B})$: $GM\_ALL(\sigma)$, where σ represents the parameter in Equation(4).

Agglomerative Clustering with $\mathcal{C}^3_{\mathbb{N}}(\mathbb{A}, \mathbb{B})$: $AM\_EM(\sigma)$, where σ represents the parameter in Equation(4).

Agglomerative Clustering with $\mathcal{C}^4_{\mathbb{N}}(\mathbb{A}, \mathbb{B})$: $GM\_EM(\sigma)$, where σ represents the parameter in Equation(4).

k-means++ Clustering: $KMPP(num)$, where *num* represents the code book size.

Ward's Clustering: $WARD(num)$, where *num* represents the code book size.

Hausdorff Clustering: $HAUS$. For both between-image (feature set) distance and between-image-set distance, Hausdorff distance is used.

Image Dataset

In the experiments, we use "the ORL database of face" and "Nister's image dataset", because we have to know the ground truth for the evaluation. The ORL database consists of 400 face images, ten different images of each of 40 distinct subjects. Nister's image dataset consists of 10200 images, four different images taken under different conditions of each 2550 objects. For the Nister's dataset, top 400 images (100 objects) are used.

Local Features

The local features used in the experiments are 64D SURF features extracted by using OpenCV 2.4.2 library.

Note that we didn't use negative images in the experiments, because of the fairness, i.e., k-means++, Ward's Clustering, and Hausdorff Clustering do not use negative images.

4.3 Parameter tuning

Except for Hausdorff clustering, all methods have parameters: proposed methods have parameter σ, k-means++ and Ward's Clustering are applied to BoF vectors that depend on the code book size *num*. These parameters are tuned so as to maximize $NMI(\mathbb{X}, \mathbb{Y})$ for given dataset.

In this paper, detailed parameter tuning results are omitted, and only the resulted parameters are shown. For ORL database, $AM\_ALL(0.01)$, $GM\_ALL(0.05)$, $AM\_EM(0.005)$, $GM\_EM(0.01)$, $KMPP(50)$ and $WARD(500)$ were the best in examined parameters. For Nister's dataset, $AM\_ALL(0.05)$, $GM\_ALL(0.05)$, $AM\_EM(0.005)$, $GM\_EM(0.01)$, $KMPP(30)$ and $WARD(50)$ were the best in examined parameters.

4.4 Comparative Results

We compared NMI scores of different clustering methods with tuned parameters described above. Figure 1 shows the graphs of NMI scores on ORL database. The horizontal and vertical axes represent the number of clusters and NMI score, respectively. From this figure, $GM\_ALL(0.05)$ and $AM\_ALL(0.01)$ are competitive and provide the better results than other methods. The best NMI score is obtained at #cluster=58. The third best method is $GM\_EM(0.01)$, and the fourth best is $WARD(500)$. Following these methods, NMI scores of $AM\_EM(0.005)$, $KMPP(50)$, and $HAUS$ degenerates in this order.

Figure 2 shows the graphs of NMI scores on Nister's image dataset. From this figure, $AM\_ALL(0.05)$ and $GM\_ALL(0.05)$ are almost competitive and perform better than other methods. The third best method is $GM\_EM(0.01)$. $WARD(50)$ and $AM\_EM(0.005)$ are almost competitive. Following these methods, NMI scores of $KMPP(30)$, and $HAUS$ degenerates in this order.

Table 1 summarizes the best NMI scores for different clustering methods on different image datasets. From this table, we can notice that AM_ALL and GM_ALL are almost competitive and perform better than other examined methods. Following these methods, GM_EM and WARD are performing better than AM_EM and k-means++ , and Hausdorff clustering is the worst.

This means that agglomerative clustering based on Equations (11) and (12) have advantages over other clustering methods. Ward's method and k-means++ have been widely applied to many clustering tasks and proven as standard clustering methods. But, the experimental results demonstrate that some commonality based clustering methods perform better than these method applied to BoF image representation.

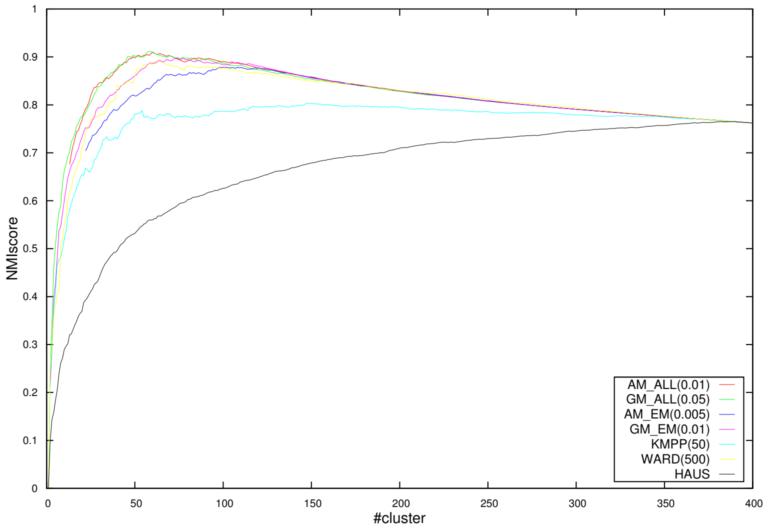


Fig. 1. NMI graphs of different clustering methods for the ORL database of faces using best parameters: Horizontal axis is the number of clusters, vertical is the NMI score.

Table 1. The NMI scores of different clustering methods on different image datasets. For the ORL database, NMI scores are measured at #cluster=40. For Nister's image dataset, NMI score is measured at #cluster=100.

dataset	clustering method						
(#clusters)	AM_ALL	GM_ALL	AM_EM	GM_EM	KMPP	WARD	HAUS
ORL(40)	.874106	**.877920**	.789015	.830801	.728934	.840150	.496100
Nister(100)	**.917861**	.911005	.841304	.879985	.823335	.844039	.729781

This implies that multiple-vector representation of images is better than BoF representation for some tasks. If this is true, there is a chance to renew Pattern Recognition or Image Retrieval algorithms for multiple-vector representation without using BoF.

4.5 Examples of Clustered Images

Figure 3 shows examples of clustered images of ORL database by using GM_ALL (0.05) at #cluster=58, where the NMI score becomes maximum. The correct clustering results (a)∼(d) demonstrate that wearing glasses, small rotation, scale change and local shading do not affect the clustering results. The incorrect results (e)∼(h) show that if there exists similar images of other subjects, that may cause mis-clustering.

Figure 4 shows examples of clustered images of Nister's dataset by using GM_ALL(0.05) at #cluster=100. The correct clustering results (a)∼(d) demonstrate that rotations, homography, and partial occlusions do not affect the clus-

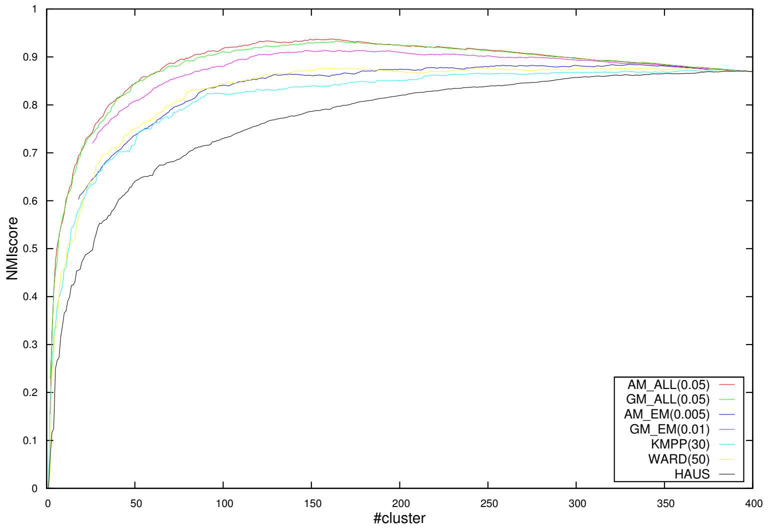


Fig. 2. NMI graphs of different clustering methods for Nister's image dataset using best parameters: Horizontal axis is the number of clusters, vertical is the NMI score.

tering results. The incorrect results (e)∼(h) show that if there exists common local features, e.g. frets and strings found in guitar and balalaika and background carpet textures may cause mis-clustering.

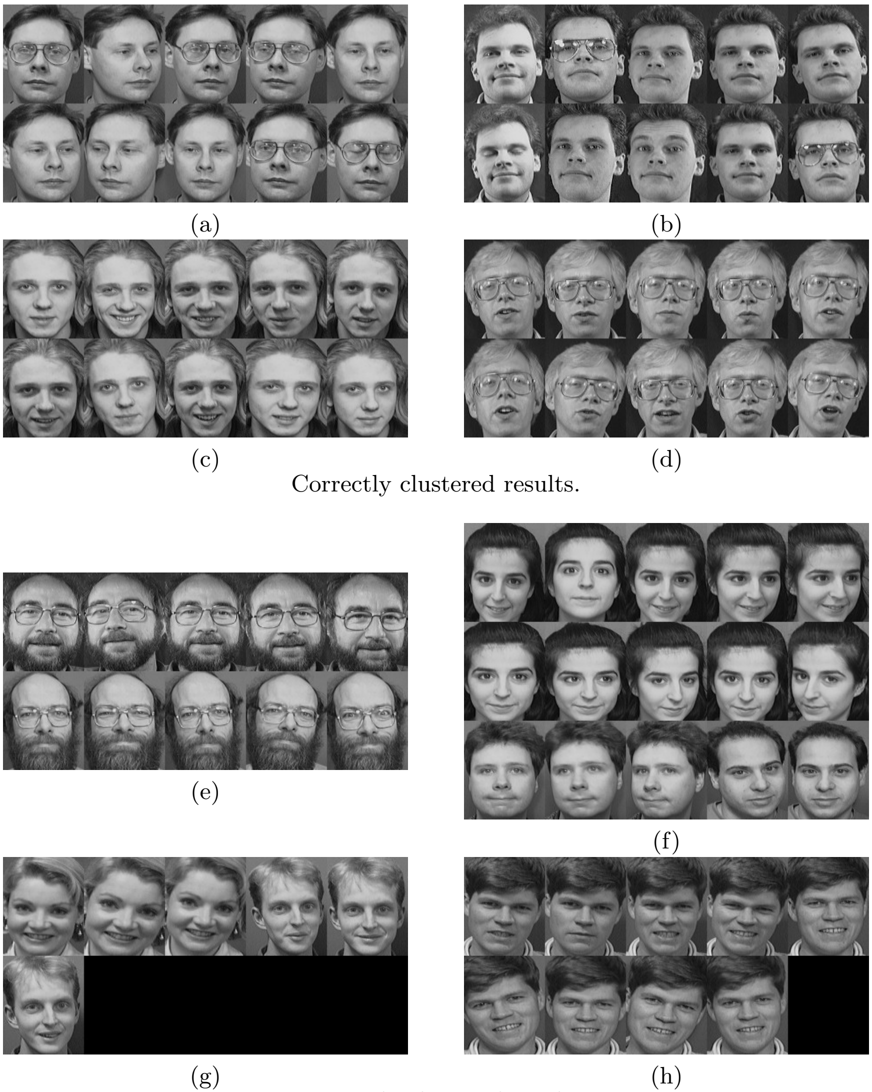
As shown above, objects having distinctive features can be clustered correctly, but objects including common local features may be clustered incorrectly. If we can properly assign negative samples, this problem can be solved or relaxed.

5 Conclusion

This paper proposes commonality preserving clustering method for local feature representation of images. Four commonality measures are proposed based on Diverse Density and examined through extensive experiments. As a result, agglomerative clustering methods using two commonality measures defined by Equations (11) and (12) are almost competitive and perform better than examined methods including k-means++, Ward's method, and Hausdorff clustering.

Essentially, similarity or dissimilarity is defined between two images, but commonality measure is naturally defined on an image set. From this viewpoint, it is clear that commonality and similarity are essentially different and the idea of commonality preserving clustering has not been proposed so far.

The commonality measure represents the number and strength of commonly existing features across the image set. The advantages of commonality preserving clustering are 1) it can be applied directly to the multiple-vector representation of images without using BoF (code book is not necessary), 2) we can guarantee



(a) (b)

(c) (d)

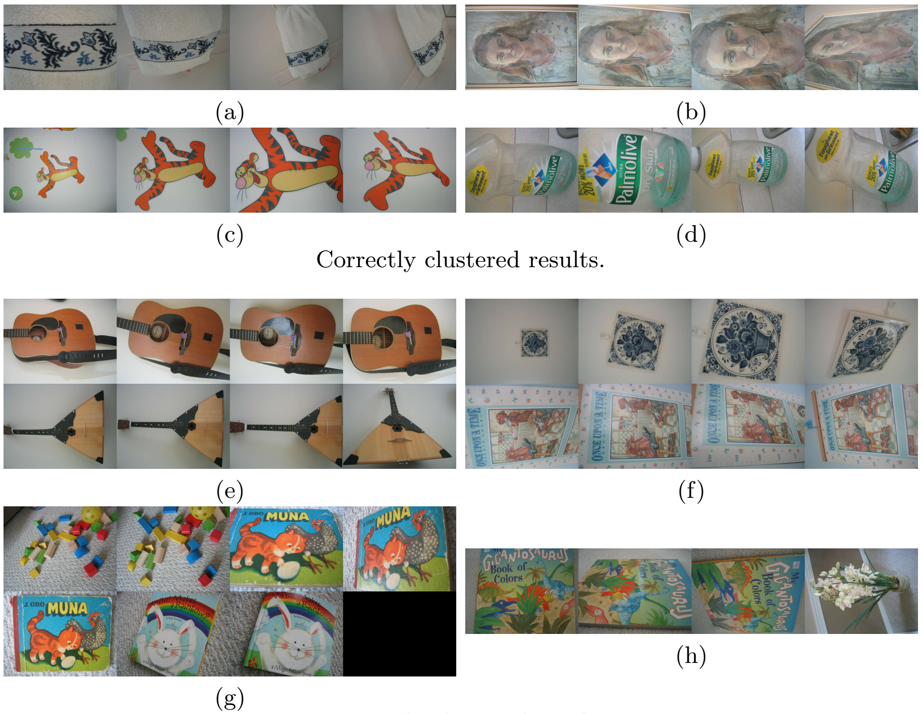
Correctly clustered results.

(e) (f)

(g) (h)

Incorrectly clustered results.

Fig. 3. Examples of clustering results on ORL database by *GM_ALL*(0.05) at #cluster=58, where NMI score becomes maximum. Correct clusterings, in other words cluster has all the images of the same subject in the dataset, are from (a) to (d), and incorrect clustering are from (e) to (h).



Correctly clustered results.

Incorrectly clustered results.

Fig. 4. Examples of clustering results on Nister's dataset by $GM\_ALL(0.05)$ at #cluster=100. Correct clusterings, in other words cluster has all the images of the same subject in the dataset, are from (a) to (d), and incorrect clustering are from (e) to (h).

the integrity of clusters in terms of common features, 3) some commonality measure produce better clustering results than other methods.

In the experiments, we didn't use negative images for guaranteeing the fairness. But, by properly assigning negative images, we can emphasize the distinctive features and enlarge the difference between clusters. Also, we can create models representing clusters by using EM-DD and the resulted models can be utilized in the classifier. These tasks should be done in the future works.

Acknowledgement. This work was supported by "R&D Program for Implementation of Anti-Crime and Anti-Terrorism Technologies for a Safe and Secure Society", Funds for integrated promotion of social system reform and research and development of the Ministry of Education, Culture, Sports, Science and Technology, the Japanese Government.

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110

2. Bay, H., Ess, A., Tiytelaars, T., Gool, L.J.V.: Surf: Speeded up robust features. CVIU **110** (2008) 346–359
3. Fei-fei, L.: A bayesian hierarchical model for learning natural scene categories. In: In CVPR. (2005) 524–531
4. Maron, O., Lozano-Pérez, T.: A framework for multiple-instance learning. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, MIT Press (1998) 570–576
5. Maron, O., Ratan, A.L.: Multiple-instance learning for natural scene classification. In: In The Fifteenth International Conference on Machine Learning, Morgan Kaufmann (1998) 341–349
6. Arthur, D., Vassilvitskii, S.: k-means++: the advantages of careful seeding. In: SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics (2007) 1027–1035
7. Ward, J.: Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association **58** (1963) 236–244
8. Berkhin, P.: A survey of clustering data mining techniques. Grouping Multidimensional Data (2006) 25–71
9. Forgy, E.: Cluster analysis of multivariate data: Efficiency versus interpretability of classification. Biometrics **21** (1965) 768–769
10. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In Cam, L.M.L., Neyman, J., eds.: Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability. Volume 1., University of California Press (1967) 281–297
11. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley & Sons (1990)
12. Jain, A., Dubes, R.: Algorithms for Clustering Data. Prentice Hall (1988)
13. Lance, G.N., Williams, W.T.: A general theory of classificatory sorting strategies 1. hierarchical systems. The Computer Journal **9** (1967) 373–380
14. Murtagh, F., Contreras, P.: Methods of hierarchical clustering. CoRR **abs/1105.0121** (2011)
15. Huttenlocher, D., Klanderman, G.A., Kl, G.A., Rucklidge, W.J.: Comparing images using the hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence **15** (1993) 850–863
16. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science **315** (2007) 972–976
17. Zhang, Q., Goldman, S.A.: Em-dd: An improved multiple-instance learning technique. In: In Advances in Neural Information Processing Systems, MIT Press (2001) 1073–1080
18. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B **39** (1977) 1–38
19. Witten, I.H., Frank, E., Holmes, G.: Data mining : practical machine learning tools and techniques. The Morgan Kaufmann series in data management systems. Morgan Kaufmann, Amsterdam, Boston, Paris (2011)