# Multi-Exemplar Affinity Propagation

Chang-Dong Wang, *Student Member*, *IEEE*, Jian-Huang Lai, *Senior Member*, *IEEE*, Ching Y. Suen, *Life Fellow*, *IEEE*, and Jun-Yong Zhu, *Student Member*, *IEEE*

**Abstract**—The affinity propagation (AP) clustering algorithm has received much attention in the past few years. AP is appealing because it is efficient, insensitive to initialization, and it produces clusters at a lower error rate than other exemplar-based methods. However, its single-exemplar model becomes inadequate when applied to model multisubclasses in some situations such as scene analysis and character recognition. To remedy this deficiency, we have extended the single-exemplar model to a multi-exemplar one to create a new multi-exemplar affinity propagation (MEAP) algorithm. This new model automatically determines the number of exemplars in each cluster associated with a super exemplar to approximate the subclasses in the category. Solving the model is NP-hard and we tackle it with the max-sum belief propagation to produce neighborhood maximum clusters, with no need to specify beforehand the number of clusters, multi-exemplars, and superexemplars. Also, utilizing the sparsity in the data, we are able to reduce substantially the computational time and storage. Experimental studies have shown MEAP's significant improvements over other algorithms on unsupervised image categorization and the clustering of handwritten digits.

**Index Terms**—Clustering, multi-exemplar, affinity propagation, factor graph, max-product belief propagation

## 1 Introduction

Affinity propagation (AP) [1] is an exemplar-based clustering method developed recently. It takes as input the similarities between data points and produces a set of exemplars and the assignments of data points to the most appropriate exemplars. The exemplars are defined to be the data points best representing the data. It utilizes the max-product belief propagation algorithm over factor graph [2] to generate clusters insensitive to initialization and converged to the neighborhood maximum [3]. As reported in [1], the AP algorithm has three advantages over other exemplar-based clustering methods: 1) It is efficient, 2) it is insensitive to initialization, and 3) it can find clusters with less error than $k$-centers (exemplar version of $k$-means [4], [5]). Therefore, it is very attractive and has been applied in many real-world applications, such as treatment portfolio design [6], ROI detection [7], tissue clustering [8], image categorization [9], subspace division [10], and so on. Meanwhile, many extensions have been developed, such as soft-constraint AP [11], Dirichlet process AP [12], streaming AP [13], semi-supervised AP [14], hierarchical AP [15], [16], and so on.


- *C.-D. Wang and J.-H. Lai are with the School of Information Science and Technology, Sun Yat-sen University, and with Guangdong Province Key Laboratory of Information Security, Waihuan East Road, Panyu District, Guangzhou, Guangdong 510006, P.R. China. E-mail: changdongwang@hotmail.com, stsljh@mail.sysu.edu.cn*
- *C.Y. Suen is with the Centre for Pattern Recognition and Machine Intelligence (CENPARMI), Concordia University, Suite EV3.403, 1445 de Maisonneuve Blvd West, Montréal, Québec H3G 1M8, Canada. E-mail: suen@cenparmi.concordia.ca.*
- *J.-Y. Zhu is with the School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou 510275, P.R. China. E-mail: jonesjunyong@gmail.com.*


*Manuscript received 20 Feb. 2011; revised 22 June 2012; accepted 11 Jan. 2013; published online 24 Jan. 2013.*
*Recommended for acceptance by C. Rother.*
*For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2011-02-0113.*
*Digital Object Identifier no. 10.1109/TPAMI.2013.28.*

Despite significant success, one drawback of AP is that it cannot model the category consisting of multiple subclasses since it represents each cluster by a single exemplar. In many applications, such as image categorization [17], face categorization [18], multifont optical character recognition [19], and handwritten digit classification [20], each category may contain several subclasses. For instance, in the natural scene categorization experiments, a scene category often contains multiple "themes" [17], for example, the street scene may contain themes like "road," "car," "pedestrian," "building," and so on. Fig. 1 illustrates two typical themes (with and without "sunset/sunrise") of the coast scene from the dataset of 13 natural scene categories (SceneClass13) [17]. Similarly, in the face categorization experiments, images of the same person in different facial expressions should be taken as in distinct subclasses [18]. In the applications of optical character recognition and handwritten digit classification, the class representing a letter or a digit could be composed of several subclasses, each corresponding to a different style or font [19], [20]. The data containing multiple subclasses obviously cannot be represented by a single exemplar, which leads to the failure of $k$-centers and AP in clustering the multiple subclasses data.

### 1.1 Previous Work

To solve the above problems, we can use the nonlinear clustering methods. Kernel-based methods [21] and spectral clustering [22] are two typical nonlinear techniques. In kernel-based clustering [21], [23], a kernel mapping is first used to embed the data into a feature space where the nonlinear pattern becomes linearly separable and then the clustering is performed in the feature space. Alternatively, in spectral clustering [22], [24], we first construct a weighted graph and then use the eigenvectors of an affinity matrix to obtain a clustering of the data. Unfortunately, as pointed out in [18] and [24], two main problems prevent the efficient use of such techniques. The first one is that it is difficult to find the appropriate kernel for each particular problem.

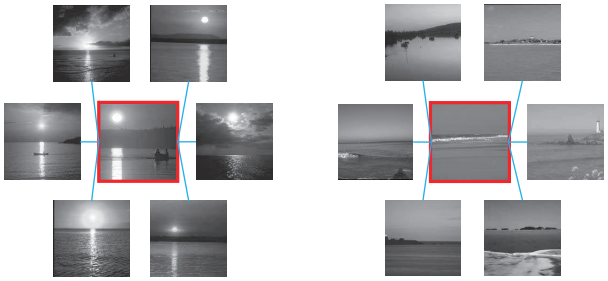


Fig. 1. Two typical themes of the coast scene: with (left) and without (right) "sunset/sunrise." Two exemplars are needed to model the scene as bounded by the red lines.

Second, the nonlinear methods usually have an associated high computational cost.

One simple but effective alternative is the multi-exemplar representation, which models a class via multiple exemplars. The multi-exemplar model can be easily found in the literatures. In supervised learning, the multi-exemplar representation was first developed in [19]. Subsequently, it has been widely used in handwritten digit classification [20], face recognition [25], and word meaning encoding [26]. Also, it has been used to enable the classifier to grow and evolve due to the change of data distribution [27]. In [28], Aiolli and Sperduti extended SVM to multiple exemplars per class so as to get very expressive decision functions without requiring the use of kernels. Another similar approach is the subclass analysis [18], where a mixture of Gaussians is used to approximate the underlying distribution of each class.

In the clustering literature, relatively less work has been done in multi-exemplar representation. The main reason is that without training samples, it is more difficult to estimate the number of clusters and subclusters used to represent each cluster. Additionally, it needs to tune more parameters than the single-exemplar representation. The first work is a hierarchical clustering method termed clustering using representatives [29], in which a constant number of well-scattered points are first chosen and then shrunk toward the cluster centroid to represent each cluster. Similarly, Liu et al. [30] developed a multiprototype clustering (MCP) algorithm for the partitional clustering in which a number of prototypes are precomputed and a separation measure is used to decide whether two precomputed prototypes should be separated or merged. Another heuristic MCP method was developed based on the minimum spanning tree [31]. Although these clustering methods can discover clusters consisting of multiple subclasses, unfortunately they have to tune some parameters. For instance, in [29], the number of points representing each cluster has to be prespecified beforehand. In [30], the thresholds deciding whether two precomputed prototypes should be separated or merged have a profound effect on the clustering results. The degree deciding whether a pattern should be taken as the potential prototype also affects the performance of the method in [31].

### 1.2 Extension of Single to Multi-Exemplar Model

Inspired by the previous work on the multi-exemplar representation and the subclass analysis, this paper extends the single-exemplar model to a multi-exemplar one and proposes a novel multi-exemplar affinity propagation (MEAP) algorithm. Each cluster is modeled by an automatically determined number of exemplars and a super-exemplar. Each data point is assigned to the most appropriate exemplar and each exemplar is assigned to the most appropriate superexemplar. The superexemplar is defined as an exemplar best representing the exemplars belonging to the corresponding cluster. The objective of the model is to maximize the sum of all similarities between data points and the corresponding exemplars *plus* the sum of all linkages between exemplars and the corresponding superexemplars. Solving the model is NP-hard. To this end, the max-sum (the $\log$-domain max-product) belief propagation [2] is utilized, producing clusters insensitive to initialization and converged to the neighborhood maximum [3]. To take advantage of the sparsity in data, we further implement a fast MEAP in which both the computational time and storage are dramatically reduced. The proposed MEAP algorithm has the following major advantages:

- It can model the category of more complex structure than AP without increasing the model complexity. Although one can learn a suitable metric for AP to characterize multisubclass structures (e.g., mapping the original pattern to an appropriate kernel space), it would make the learning model complicated and time-consuming. The MEAP algorithm inherits the advantages of AP, i.e., the same computational complexity and convergence property, but can model more complex structures.
- Compared with existing multi-exemplar clustering methods, the proposed MEAP algorithm can automatically estimate the number of clusters and the appropriate number of subclusters within each cluster. Additionally, it does not have to tune any parameter to realize this automatic estimation. On the contrary, its counterparts work well only when all the parameters are properly adjusted.

The remainder of this paper is organized as follows: Section 2 briefly introduces the key elements of AP. In Section 3, we describe the new MEAP method. The multi-exemplar model is first described and its underlying rationale is discussed. Then the max-sum belief propagation-based optimization for the model is introduced. We also implement a Fast MEAP to take advantage of the sparsity in data. The theoretical comparison between AP and MEAP is conducted to show that the AP algorithm can be viewed as a special case of MEAP. Sections 4 and 5 report the experimental results on three image categorization datasets and two handwritten digit datasets, respectively. Section 6 concludes this paper.

## 2 Affinity Propagation

AP is a single-exemplar clustering algorithm using max-sum (max-product) belief propagation to obtain good exemplars. Given a user-defined similarity matrix $[s_{ij}]_{N\times N}$ of $N$ points, it aims at searching for a valid configuration of labels $\mathbf{c} = [c_1, \ldots, c_N]$ to maximize the following objective function [32]:

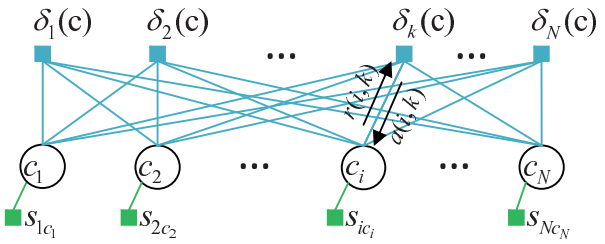



Fig. 2. Factor graph of the AP method and its messages.

$$\mathcal{S}(\mathbf{c}) = \sum_{i=1}^{N} s_{ici} + \sum_{k=1}^{N} \delta_k(\mathbf{c}), \tag{1}$$

where $\delta_k(\mathbf{c})$ is an *exemplar-consistency* constraint such that if some data point $i$ has selected $k$ as its exemplar, i.e., $c_i = k$, then data point $k$ must select itself as an exemplar, i.e., $c_k = k$,

$$\delta_k(\mathbf{c}) = \begin{cases} -\infty, & \text{if } c_k \neq k \text{ but } \exists i : c_i = k, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

AP is an optimized max-sum belief propagation algorithm over the factor graph in Fig. 2. It begins by simultaneously considering all data points as potential exemplars, and recursively transmits real-valued messages between data points until high-quality exemplars emerge. There are two kinds of messages, which are $r(i,k)$, sent from point $i$ to the candidate exemplar $k$, reflecting the accumulated evidence for how well-suited point $k$ is to serve as the exemplar for point $i$, and $a(i,k)$, sent from the candidate exemplar $k$ to point $i$, reflecting the accumulated evidence for how appropriate it would be for point $i$ to choose point $k$ as its exemplar (see Fig. 2). They are initialized as zero and updated, respectively, as follows:

$$\begin{aligned} r(i,k) &\leftarrow s_{ik} - \max_{j \neq k}[s_{ij} + a(i,j)], \\ a(i,k) &\leftarrow \min\left[0, r(k,k) + \sum_{i' \notin \{i,k\}} \max[0, r(i',k)]\right], k \neq i, \\ a(k,k) &\leftarrow \sum_{i' \neq k} \max[0, r(i',k)]. \end{aligned}$$

The assignment vector $\mathbf{c} = [c_1, \ldots, c_N]$ is computed as $c_i = \arg\max_j [a(i,j) + r(i,j)]$ after convergence.

## 3 Multi-Exemplar Affinity Propagation

Let $[s_{ij}]_{N\times N}$ be a user-defined similarity matrix with $s_{ij}$ measuring the similarity between point $i$ and the potential exemplar $j$, and $[l_{ij}]_{N\times N}$ a linkage matrix with $l_{ij}$ measuring the linkage between exemplar $i$ and its potential superexemplar $j$. We develop a multi-exemplar model that seeks two mappings, $\psi_1 : \{1, \ldots, N\} \rightarrow \{1, \ldots, N\}$ assigning data point $i$ to exemplar $\psi_1(i)$, and $\psi_2 : \{\psi_1(1), \ldots, \psi_1(N)\} \rightarrow \{\psi_1(1), \ldots, \psi_1(N)\}$ assigning exemplar $\psi_1(i)$ to superexemplar $\psi_2(\psi_1(i)) = (\psi_2 \circ \psi_1)(i)$, where $\circ$ denotes the function composition. The goal is to maximize the sum $\mathcal{S}_1$ of all similarities between data points and the corresponding exemplars *plus* the sum $\mathcal{S}_2$ of all linkages between exemplars and the corresponding superexemplars.

### 3.1 The Model

Let $C = [c_{ij}]_{N\times N}$ be an assignment matrix, where the nondiagonal elements $c_{ij} \in \{0, 1\}$ $(j \neq i)$ denote that point $j$ is the exemplar of point $i$ if $c_{ij} = 1$, i.e., $\psi_1(i) = j$, and the diagonal elements $c_{ii} \in \{0, \ldots, N\}$ denote that exemplar $c_{ii}$ is the superexemplar of exemplar $i$ if $c_{ii} \in \{1, \ldots, N\}$, i.e., $\psi_2(i) = c_{ii}$,

$$c_{ij} = \begin{cases} 1, & \text{if } j \text{ is an exemplar of } i \\ 0, & \text{otherwise,} \end{cases} \quad \forall i \neq j, \tag{3}$$

$$c_{ii} = \begin{cases} k \in \{1, \ldots, N\}, & \text{if } k \text{ is a superexemplar of } i, \\ 0, & \text{if } i \text{ is not an exemplar.} \end{cases} \tag{4}$$

The sum of all similarities between data points and the corresponding exemplars, i.e., $\mathcal{S}_1$, and the sum of all linkages between exemplars and the corresponding superexemplars, i.e., $\mathcal{S}_2$, can be, respectively, expressed as

$$\mathcal{S}_1 = \sum_{i=1}^{N} \sum_{j=1}^{N} s_{ij} \cdot [c_{ij} \neq 0], \mathcal{S}_2 = \sum_{i=1}^{N} l_{ic_{ii}} \cdot [c_{ii} \neq 0], \tag{5}$$

where $[\cdot]$ is the Iverson notation with $[\text{true}] = 1$ and $[\text{false}] = 0$. We define a function matrix $[S_{ij}(c_{ij})]_{N\times N}$ with nondiagonal elements incorporating the similarities $s_{ij}$ between data point $i$ and the potential exemplar $j$, and diagonal elements incorporating the exemplar preference $s_{ii}$ *plus* the linkage $l_{ic_{ii}}$ between exemplar $i$ and its superexemplar $c_{ii}$. That is,

$$S_{ij}(c_{ij}) = \begin{cases} s_{ij}, & \text{if } i \neq j \ \& \ c_{ij} \neq 0, \\ s_{ii} + l_{ic_{ii}}, & \text{if } i = j \ \& \ c_{ii} \neq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

We have $\mathcal{S}_1 + \mathcal{S}_2 = \sum_{i=1}^{N} \sum_{j=1}^{N} S_{ij}(c_{ij})$. The valid assignment matrix $C$ must satisfy the following three constraints:

1. *Exemplar's "1-of-N"*constraint [33]. Each data point $i$ must be assigned to *exactly one* exemplar:

$$I_i(c_{i1}, \ldots, c_{iN}) = \begin{cases} -\infty, & \text{if } \sum_{j=1}^{N} [c_{ij} \neq 0] \neq 1, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

2. *Exemplar consistency* constraint [33]. If there exists a data point $i$ selecting data point $j$ as its exemplar, then data point $j$ must be an exemplar itself:

$$E_j(c_{1j}, \ldots, c_{Nj}) = \begin{cases} -\infty, & \text{if } c_{jj} = 0 \text{ but } \exists i : c_{ij} = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

3. *Superexemplar consistency* constraint. If some exemplar $i$ has chosen exemplar $k$ as its superexemplar, i.e., $c_{ii} = k$, then $k$ must be a superexemplar itself:

$$F_k(c_{11}, \ldots, c_{NN}) = \begin{cases} -\infty, & \text{if } c_{kk} \neq k \text{ but } \exists i : c_{ii} = k, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

The goal of the multi-exemplar model is to maximize the following objective function:

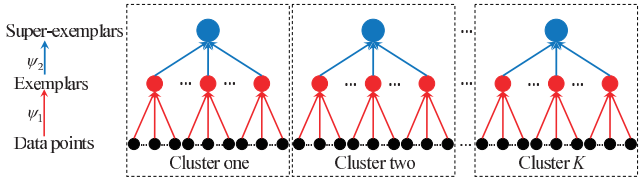



Fig. 3. Illustration of the multi-exemplar model. The mapping $\psi_1$ assigns each data point to the most appropriate exemplar and the mapping $\psi_2$ assigns each exemplar to the most appropriate superexemplar. This model can effectively characterize clusters consisting of multiple subclusters.

$$\begin{aligned}\mathcal{S}(C) &= \mathcal{S}_1 + \mathcal{S}_2 + \text{three constraints} \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N} S_{ij}(c_{ij}) + \sum_{i=1}^{N} I_i(c_{i1},\ldots,c_{iN}) \\ &\quad + \sum_{j=1}^{N} E_j(c_{1j},\ldots,c_{Nj}) + \sum_{k=1}^{N} F_k(c_{11},\ldots,c_{NN}).\end{aligned} \tag{10}$$

Fig. 3 illustrates the multi-exemplar model. The data points, exemplars, and superexemplars form a two-layer structure by the two mappings $\psi_1$ and $\psi_2$. The lower layer is modeled by the mapping $\psi_1$. The sum of all similarities between data points and the corresponding exemplars, i.e., $\mathcal{S}_1$, is used to measure the *within-subcluster* compactness. The higher layer is modeled by the mapping $\psi_2$. The sum of all linkages between exemplars and the corresponding superexemplars, i.e., $\mathcal{S}_2$, is used to measure the *within-cluster* compactness. From the single-exemplar theory, maximizing the within-cluster similarity automatically maximizes the between-cluster separation [34]. Therefore, the appropriate multi-exemplar model should be that both the *within-subcluster* compactness and the *within-cluster* compactness are maximized. Maximizing $\mathcal{S}_1 + \mathcal{S}_2$ under the constraints of producing valid clusters (i.e., $I, E, F$) makes the model effectively characterize clusters consisting of multiple subclusters. Fig. 4 compares MEAP with AP in one synthetic dataset. From the viewpoint of the maximum margin clustering [35], MEAP finds better decision boundaries than AP, i.e., larger margins are obtained.

## 3.2 Optimization

Exactly searching for an optimal assignment matrix that maximizes the objective function (10) is NP-hard since the multi-exemplar model is a generalization of the single-exemplar one, the optimization of which is proven to be NP-hard [1]. To this end, the max-sum belief propagation is utilized, which is a local-message-passing algorithm guaranteed to converge to the neighborhood maximum [3]. The factor graph is shown in Fig. 5. There are seven types of messages passing between variable nodes and function nodes, as shown in Fig. 6. In the max-sum algorithm, the message updating involves either a message from a variable to each adjacent function or that from a function to each adjacent variable. The message from a variable to a function sums together the messages from all adjacent functions except the one receiving the message [2]:

$$\mu_{x\to f}(x) \leftarrow \sum_{h\in \mathrm{ne}(x)\setminus\{f\}} \mu_{h\to x}(x), \tag{11}$$

where $\mathrm{ne}(x)$ denotes the set of adjacent functions of variable $x$. The message from a function to a variable involves a maximization over all arguments of the function except the variable receiving the message [2]:

$$\mu_{f\to x}(x) \leftarrow \max_{X\setminus\{x\}}\left[f(X) + \sum_{y\in X\setminus\{x\}} \mu_{y\to f}(y)\right], \tag{12}$$

where $X = \mathrm{ne}(f)$ is the set of arguments of function $f$. Since the messages associated with the nondiagonal variables (i.e., $c_{ij}, i \neq j$) and the diagonal variables (i.e., $c_{ii}$) are quite different, we will discuss them separately.

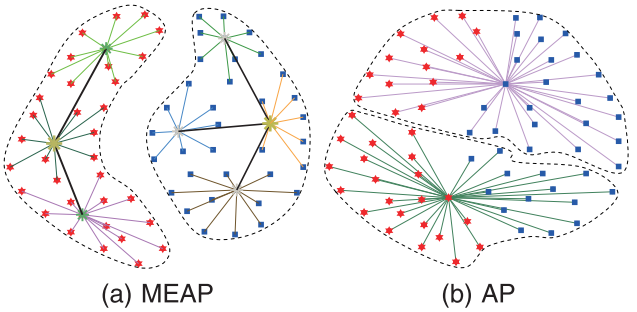



Fig. 4. MEAP versus AP. The two ground-truth categories are plotted by red hexagrams and blue squares, respectively. The decision boundaries are plotted by dash curves. Each point is assigned to the most appropriate exemplar by thin lines. In MEAP, each exemplar (small "⋆") is assigned to its most appropriate superexemplar (large "⋆") by thick lines.

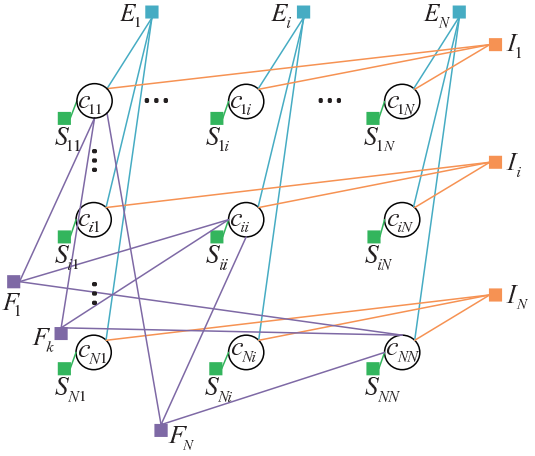



Fig. 5. Factor graph of MEAP.

### 3.2.1 Messages of Nondiagonal Elements

As shown on the left of Fig. 6, there are five types of messages associated with $c_{ij}, i \neq j$ as follows ($m = 0, 1$):

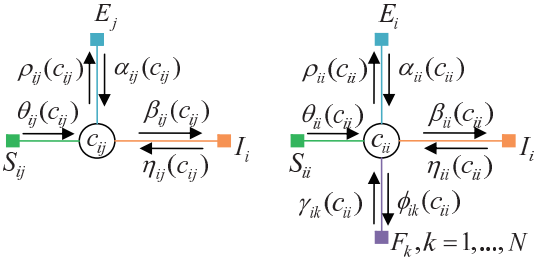



Fig. 6. Messages of MEAP.

$$\theta_{ij}(m) = \mu_{S_{ij}\to c_{ij}}(m) = S_{ij}(m), \tag{13}$$

$$\rho_{ij}(m) = \mu_{c_{ij}\to E_j}(m) = \theta_{ij}(m) + \eta_{ij}(m), \tag{14}$$

$$\begin{aligned}\alpha_{ij}(m) &= \mu_{E_j\to c_{ij}}(m)\\ &= \max_{c_{i'j}:i'\neq i}\Big[E_j(c_{1j},\ldots,c_{Nj}) + \sum_{i''\neq i}\rho_{i''j}(c_{i''j})\Big],\end{aligned} \tag{15}$$

$$\beta_{ij}(m) = \mu_{c_{ij}\to I_i}(m) = \theta_{ij}(m) + \alpha_{ij}(m), \tag{16}$$

$$\begin{aligned}\eta_{ij}(m) = \mu_{I_i\to c_{ij}}(m) = \max_{c_{ij'}:j'\neq j}[I_i(c_{i1},\ldots,c_{iN})\\ + \sum_{j''\neq j}\beta_{ij''}(c_{ij''})].\end{aligned} \tag{17}$$

According to (6) and (13), these messages take as input the similarity $s_{ij}$, $i \neq j$. Consequently, these messages reflect the accumulated evidence for deciding the partial mapping $\psi_1$ from data point $i$ to the potential exemplar $j$ ($j \neq i$). That is, $m = c_{ij} = 1$ implies that $\psi_1(i) = j$, and $m = c_{ij} = 0$ implies that $\psi_1(i) \neq j$.

#### 3.2.2 Messages of Diagonal Elements

There are seven types of messages associated with $c_{ij}$, $i = j$ (the right of Fig. 6) as follows ($m = 0,\ldots,N$):

$$\theta_{ii}(m) = \mu_{S_{ii}\to c_{ii}}(m) = S_{ii}(m), \tag{18}$$

$$\rho_{ii}(m) = \mu_{c_{ii}\to E_i}(m) = \theta_{ii}(m) + \eta_{ii}(m) + \sum_{k=1}^{N}\gamma_{ik}(m), \tag{19}$$

$$\begin{aligned}\alpha_{ii}(m) &= \mu_{E_i\to c_{ii}}(m)\\ &= \max_{c_{i'i}:i'\neq i}[E_i(c_{1i},\ldots,c_{Ni}) + \sum_{i''\neq i}\rho_{i''i}(c_{i''i})],\end{aligned} \tag{20}$$

$$\beta_{ii}(m) = \mu_{c_{ii}\to I_i}(m) = \theta_{ii}(m) + \alpha_{ii}(m) + \sum_{k=1}^{N}\gamma_{ik}(m), \tag{21}$$

$$\begin{aligned}\eta_{ii}(m) &= \mu_{I_i\to c_{ii}}(m)\\ &= \max_{c_{ii'}:i'\neq i}[I_i(c_{i1},\ldots,c_{iN}) + \sum_{i''\neq i}\beta_{ii''}(c_{ii''})],\end{aligned} \tag{22}$$

$$\begin{aligned}\phi_{ik}(m) = \mu_{c_{ii}\to F_k}(m) = \theta_{ii}(m) + \alpha_{ii}(m)\\ + \eta_{ii}(m) + \sum_{k'\neq k}\gamma_{ik'}(m),\end{aligned} \tag{23}$$

$$\begin{aligned}\gamma_{ik}(m) &= \mu_{F_k\to c_{ii}}(m)\\ &= \max_{c_{i'i'}:i'\neq i}[F_k(c_{11},\ldots,c_{NN}) + \sum_{j\neq i}\phi_{jk}(c_{jj})].\end{aligned} \tag{24}$$

According to (6) and (18), these messages take as input both of the exemplar preference $s_{ii}$ and the linkage $l_{ic_{ii}}$ between exemplars and superexemplars. Consequently, these messages reflect the accumulated evidence for simultaneously deciding the partial mapping $\psi_1$ from point $i$ to itself and the mapping $\psi_2$ from exemplar $i$ to the potential superexemplar $k$ ($k$ can be either $k = i$ or $k \neq i$). That is, $m = c_{ii} \in \{1,\ldots,N\}$ implies that $\psi_1(i) = i, \psi_2(i) = c_{ii}$, and $m = c_{ii} = 0$ implies that $\psi_1(i) \neq i$ and $i$ is not in the domain of $\psi_2$.

#### 3.2.3 Simplified Messages

By applying some mathematical tricks used in [1] and [33], we can obtain the simplified messages as follows:

$$\begin{aligned}&i \neq j\\ &\tilde{\rho}_{ij} \leftarrow s_{ij} - \max\Big[\max_{j'\notin\{j,i\}}\big[s_{ij'} + \tilde{\alpha}_{ij'}\big],\\ &\qquad\max_{m\in\{1,\ldots,N\}}[l_{im} + \tilde{\gamma}_{im}] + s_{ii} + \tilde{\alpha}_{ii}\Big],\end{aligned} \tag{25}$$

$$\tilde{\alpha}_{ij} \leftarrow \min\Big[0, \max_{m\in\{1,\ldots,N\}}\tilde{\rho}_j^m + \sum_{i'\notin\{i,j\}}\max[0,\tilde{\rho}_{i'j}]\Big], \tag{26}$$

$$\begin{aligned}&\forall i = 1,\ldots,N, k = 1,\ldots,N\\ &\tilde{\rho}_i^k \leftarrow s_{ii} + l_{ik} - \max_{i'\neq i}[s_{ii'} + \tilde{\alpha}_{ii'}] + \tilde{\gamma}_{ik},\end{aligned} \tag{27}$$

$$\tilde{\alpha}_{ii} \leftarrow \sum_{i'\neq i}\max\big[0,\tilde{\rho}_{i'i}\big], \tag{28}$$

$$\tilde{\phi}_{ik} \leftarrow \min\Big[l_{ik} - \max_{m\neq k}[l_{im} + \tilde{\gamma}_{im}], \tilde{\alpha}_{ii} + \tilde{\rho}_i^k - \tilde{\gamma}_{ik}\Big], \tag{29}$$

$$\tilde{\gamma}_{kk} \leftarrow \sum_{i'\neq i}\max\big[0,\tilde{\phi}_{i'k}\big], \tag{30}$$

$$\tilde{\gamma}_{ik} \leftarrow \min\Big[0, \tilde{\phi}_{kk} + \sum_{i'\notin\{i,k\}}\max[0,\tilde{\phi}_{i'k}]\Big], k \neq i, \tag{31}$$

where

$$\begin{aligned}\tilde{\rho}_{ij} &= \rho_{ij}(1) - \rho_{ij}(0), \tilde{\alpha}_{ij} = \alpha_{ij}(1) - \alpha_{ij}(0), \tilde{\rho}_i^k\\ &= \rho_{ii}(k) - \rho_{ii}(0), \tilde{\alpha}_{ii} = \alpha_{ii}(k) - \alpha_{ii}(0), \tilde{\phi}_{ik}\\ &= \phi_{ik}(k) - \max_{m\neq k}\phi_{ik}(m), \tilde{\gamma}_{ik}\\ &= \gamma_{ik}(k) - \gamma_{ik}(m : m \neq k),\end{aligned}$$

and $\beta, \eta$ are eliminated. The values of all messages $\tilde{\rho}$, $\tilde{\alpha}$, $\tilde{\phi}$, and $\tilde{\gamma}$ are initialized as zero and updated via (25) to (31). The derivation can be found in the supplemental material, which can be found in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2013.28. The message-updating procedure may be terminated after the local decisions stay constant for some number of iterations $t_{conv}$ or after a fixed number of iterations $t_{max}$.

#### 3.2.4 Computing Assignment Matrix

To estimate the value of an element $c_{ij}$, we sum together all incoming messages to $c_{ij}$ and take the value $\hat{c}_{ij}$ that maximizes the sum. That is,

$$\begin{aligned}\hat{c}_{ij} &= \arg\max_{c_{ij}} \left[\theta_{ij}(c_{ij}) + \alpha_{ij}(c_{ij}) + \eta_{ij}(c_{ij})\right], \\ &= \begin{cases} 1, & \text{if } \tilde{\alpha}_{ij} + \tilde{\rho}_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases} \forall i \neq j, \end{aligned} \tag{32}$$

$$\begin{aligned}\hat{c}_{ii} &= \arg\max_{c_{ii}} \left[\theta_{ii}(c_{ii}) + \alpha_{ii}(c_{ii}) + \eta_{ii}(c_{ii}) + \sum_{k=1}^{N} \gamma_{ik}(c_{ii})\right], \\ &= \begin{cases} \arg\max_k \tilde{\rho}_i^k, & \text{if } \tilde{\alpha}_{ii} + \max_k \tilde{\rho}_i^k \geq 0, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{33}$$

From the assignment matrix, we obtain the two mappings $\psi_1$ and $\psi_2$, and thus generate the clustering labels $\{(\psi_2 \circ \psi_1)(1), \ldots, (\psi_2 \circ \psi_1)(N)\}$ of $N$ data points.

### 3.3 Sparse Similarity and Fast MEAP

Like AP, MEAP is well suited to taking advantage of the sparsity in data.[1] As discussed in [32], the computational complexity of the message passing through each iteration is $\mathcal{O}(N^2)$, which takes the $\sum$, $\max$, and $\min$ operations as a single step. In some applications [1], the problems are structured in such a way that many data points cannot be represented by some others as exemplars and superexemplars. That is, $\exists i, j$ such that $s_{ij} = -\infty, l_{ij} = -\infty$. In this circumstance, $\tilde{\rho}_{ij}$ is automatically $-\infty$ according to (25). The direct result is that we do not need to compute $\tilde{\rho}_{ij}$ and can simplify any computation involving $\tilde{\rho}_{ij}$ by eliminating the corresponding term, such as the sum of $\max[0, \tilde{\rho}_{ij}]$ in (26) and (28). Additionally, we do not need to compute the inverse message of $\tilde{\rho}_{ij}$, namely, $\tilde{\alpha}_{ij}$, since $\tilde{\alpha}_{ij}$ becomes inconsequential. That is, the computations of $\max[s_{ij} + \tilde{\alpha}_{ij}]$ in (25) and (27) and the assignment matrix $\tilde{\alpha}_{ij} + \tilde{\rho}_{ij} > 0$ in (32) are overwhelmed by $s_{ij}$ and $\tilde{\rho}_{ij}$, respectively, which are $-\infty$. Similarly, $\tilde{\rho}_i^j$ is automatically $-\infty$ according to (27) since $l_{ij} = -\infty$. $\phi_{ij}$ is also equal to $-\infty$ according to (29) since $l_{ij} = -\infty, \tilde{\rho}_i^j = -\infty$ and $\min(-\infty, -\infty) = -\infty$. The direct result is that we do not need to compute $\tilde{\rho}_i^j$ and $\tilde{\phi}_{ij}$. If we investigate the computations involving $\tilde{\gamma}_i^j$, we find that $\tilde{\gamma}_i^j$ also does not need to be computed since $\tilde{\gamma}_i^j$ is only used in computing $\tilde{\rho}_i^j$ and $\tilde{\phi}_{ij}$.

Given a sparse dataset with $N$ data points but only $M$ ($M < N^2$) pairs of $(i, j) \in \{1, \ldots, N\}^2$ such that $s_{ij} > -\infty$, $l_{ij} > -\infty$. Let $\Omega \subset \{1, \ldots, N\}^2$ denote the $M$ pairs of $(i, j)$ satisfying $s_{ij} > -\infty, l_{ij} > -\infty$. We can rewrite the messages from (25) to (31) as follows and call it Fast MEAP:

$$\begin{aligned} & i \neq j \& (i, j) \in \Omega \\ \tilde{\rho}_{ij} \quad \leftarrow \quad & s_{ij} - \max\Big[\max_{\substack{j' \notin \{j,i\} \\ (i,j') \in \Omega}} [s_{ij'} + \tilde{\alpha}_{ij'}], \\ & \max_{m:(i,m)\in\Omega} [l_{im} + \tilde{\gamma}_{im}] + s_{ii} + \tilde{\alpha}_{ii}\Big], \end{aligned} \tag{34}$$

$$\tilde{\alpha}_{ij} \leftarrow \min\left[0, \max_{m:(j,m)\in\Omega} \tilde{\rho}_j^m + \sum_{\substack{i' \notin \{i,j\} \\ (i',j) \in \Omega}} \max[0, \tilde{\rho}_{i'j}]\right], \tag{35}$$

$$\begin{aligned} & \forall i = 1, \ldots, N, \forall k = 1, \ldots, N, \text{and} \forall (i, k) \in \Omega \\ & \tilde{\rho}_i^k \leftarrow s_{ii} + l_{ik} - \max_{\substack{i' \neq i \\ (i,i') \in \Omega}} [s_{ii'} + \tilde{\alpha}_{ii'}] + \tilde{\gamma}_{ik}, \end{aligned} \tag{36}$$

$$\tilde{\alpha}_{ii} \leftarrow \sum_{\substack{i' \neq i \\ (i',i) \in \Omega}} \max\left[0, \tilde{\rho}_{i'i}\right], \tag{37}$$

$$\tilde{\phi}_{ik} \leftarrow \min\left[l_{ik} - \max_{\substack{m \neq k \\ (i,m) \in \Omega}} [l_{im} + \tilde{\gamma}_{im}], \tilde{\alpha}_{ii} + \tilde{\rho}_i^k - \tilde{\gamma}_{ik}\right], \tag{38}$$

$$\tilde{\gamma}_{kk} \leftarrow \sum_{\substack{i' \neq i \\ (i',k) \in \Omega}} \max\left[0, \tilde{\phi}_{i'k}\right], \tag{39}$$

$$\tilde{\gamma}_{ik} \leftarrow \min\left[0, \tilde{\phi}_{kk} + \sum_{\substack{i' \notin \{i,k\} \\ (i',k) \in \Omega}} \max[0, \tilde{\phi}_{i'k}]\right], k \neq i. \tag{40}$$

In terms of storage, the sparse structure can be stored for quick traversal using $2M$ integers for the $M$ pairs and $1M$ floating-point values for the similarity, rather than $N^2$ floating-point values; and all the messages are stored as $5 \times M + 2 \times N$ floating-point values, rather than $5 \times N^2$ floating-point values. In terms of computational complexity, only $5 \times M + 2 \times N$ messages need to be computed and exchanged, and the computation can be more efficient since there are fewer terms in the $\sum$, $\max$, and $\min$ operations in each message.

### 3.4 Comparison to Affinity Propagation

The proposed MEAP algorithm can be viewed as a generalization of AP. The single-exemplar margin is a lower bound on the margin for the multi-exemplar model since it can be obtained when all the exemplars belonging to the same cluster coincide, or that setting the linkage matrix $[l_{ij}]_{N \times N} = 0$. Both of them can be implemented to take advantage of the sparsity in data. Additionally, they have the same computational complexity, which can be revealed by comparing their simplified messages. That is, the number of messages of MEAP is seven while the number of messages of AP is three and all the messages are of the same computational complexity.

They both do not require preselecting the number of clusters and initial exemplars/superexemplars by initially considering all data points as the potential exemplars/superexemplars, i.e., initializing all messages as zero. Similarly to AP, setting the exemplar preference $s_{ii}$ to a higher value would generate a larger number of exemplars, yielding a more detailed description of the subcluster structure within each cluster and vice versa. Setting the superexemplar preference $l_{ii}$ to a higher value would generate a larger number of superexemplars and vice versa. Therefore, when the cluster number is unknown in advance and there is no prior knowledge of the complexity of subcluster structures, the exemplar preference $s_{ii}$ and superexemplar preference $l_{ii}$ should be, respectively, set to median values of the similarity and linkage matrices. This may generate a moderate number of subclusters per cluster and a moderate number of clusters.

1. The data is sparse in case the matrix of similarities has zero entries; in the log-domain the missing similarities become $-\infty$.

Like AP, the main cause of failure mode of MEAP is that the objective function (10) has multiple minima with corresponding multiple fixed points of the update rules, which may prevent convergence. In this case, the message update may oscillate, with data points alternating between being exemplars and nonexemplars and exemplars alternating between being superexemplars and nonsuperexemplars. One remedy is to introduce a damping to the message update, which could always avoid oscillations. Let $\mu$ denote any of the seven messages on the left-hand side of equations (25) to (31); the damping is done as follows [32]:

$$\mu = \lambda\mu^{\text{old}} + (1-\lambda)\mu^{\text{new}}. \tag{41}$$

The higher values of the damping factor $\lambda$ unsurprisingly lead to slower convergence rates but often lead to more stable maximization (i.e., avoiding oscillations). According to experimental results, setting the damping factor $\lambda$ to 0.9 is sufficient in most cases to ensure convergence [32].

## 4 Unsupervised Image Categorization

In this section, we investigate the improvement of MEAP w.r.t. AP in unsupervised image categorization on three commonly tested image datasets, which are the Japanese female facial expression database (JAFFE) [36], the Caltech101 dataset [37], and the 13 natural scene categories dataset (SceneClass13) [17]. Some existing clustering methods have been performed and compared with MEAP, including the classical $k$-centers, kernel-based clustering, spectral clustering, MCP, and hierarchical AP. Comparative results have shown that MEAP significantly outperforms AP and is comparable with state-of-the-art clustering methods.

### 4.1 Methods and Settings

The compared methods and their parameter settings are summarized as follows:

1. *Exemplar-based clustering methods.* They are $k$-centers [4] and AP [1]. The parameters for AP and MEAP are set as follows: $t_{conv} = 100$, $t_{max} = 1{,}000$, $\lambda = 0.9$ as in [1] and [9]. The maximum number of iterations is set to 1,000 and in each run a random initialization is used in $k$-centers.
2. *Kernel-based clustering.* The compared kernel clustering methods include kernel $k$-means (k$k$-means) [21] and conscience online learning (COLL) [23]. Gaussian kernel $\kappa(\mathbf{x}_i, \mathbf{x}_j) = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2}$ is used to construct the kernel matrix $K$. To obtain a meaningful comparison, on each dataset, the most appropriate $\sigma$ obtained by the criterion proposed in [38] is used.
3. *Spectral clustering.* The widely used normalized cut (Ncut) [22] and recently proposed Graclus [24] are performed. Graphs are constructed the same as in [22], and in Graclus the ratio association is used.
4. *MPC.* The multiprototype clustering proposed in [30] is performed and compared.
5. *Hierarchical AP.* The greedy hierarchical affinity propagation (GHAP) [15] and a theoretically improved version HAP [16] are performed. The layer number is set to 2.

All the experiments are implemented in Matlab7.8.0.347 (R2009a) 64-bit edition on a workstation (Windows 64 bit, 8 Intel 2.00 GHz processors, 16 GB of RAM).

### 4.2 Similarity Metric

Previous works [9], [17], [39], [40] utilized SIFT features [41] for image matching and category learning. Each feature is described by a 128D vector. We follow the procedure described by Lowe to count the number of significant feature matches comparing image $i$ with image $k$ (denoted as $\mathcal{M}_{ik}$): For each local feature from image $i$, the nearest and second nearest features are sought in image $k$. The match is considered significant if the distance ratio between the nearest and second-nearest neighbors is greater than a threshold $\zeta$ which is selected from 0.5 to 0.9 [41]. Then the similarity matrix $[s_{ij}]_{N\times N}$ is defined to be the number of significant feature matches normalized by subtracting means across both dimensions [9]:

$$s_{ij} = \mathcal{M}_{ij} - \frac{1}{N}\sum_{k=1}^{N}\mathcal{M}_{ik} - \frac{1}{N}\sum_{k=1}^{N}\mathcal{M}_{kj}. \tag{42}$$

The linkage matrix $[l_{ij}]_{N\times N}$ is defined as $l_{ij} = s_{ij}/N$. The exemplar preference $s_{ii}$ (and, in consequence, the superexemplar preference) is adjusted over a range of values to produce different numbers of clusters.

### 4.3 Clustering Evaluations

Two widely adopted external evaluations, namely, normalized mutual information (NMI) [42] and classification rate (CR), are used in measuring how closely the clustering and underlying class labels match. Although there exist many external clustering evaluation measurements, such as clustering errors, average purity, entropy-based measures [43], and pair counting-based indices [44], the mutual information provides a sound indication of the shared information between a pair of clusterings [42], [45].

Given a dataset $\mathcal{X}$ of size $n$, the clustering labels $\pi$ of $c$ clusters, and actual class labels $\zeta$ of $\hat{c}$ classes, a confusion matrix is formed first, where entry $(i, j)$, $n_i^{(j)}$ gives the number of points in cluster $i$ and class $j$. Then NMI can be computed from the confusion matrix [42]:

$$NMI = \frac{2\sum_{l=1}^{c}\sum_{h=1}^{\hat{c}}\frac{n_l^{(h)}}{n}\log\frac{n_l^{(h)}n}{\sum_{i=1}^{c}n_i^{(h)}\sum_{i=1}^{\hat{c}}n_l^{(i)}}}{H(\pi)+H(\zeta)}, \tag{43}$$

where $H(\pi) = -\sum_{i=1}^{c}\frac{n_i}{n}\log\frac{n_i}{n}$ and $H(\zeta) = -\sum_{j=1}^{\hat{c}}\frac{n^{(j)}}{n}\log\frac{n^{(j)}}{n}$ are the Shannon entropy of cluster labels $\pi$ and class labels $\zeta$, respectively, with $n_i$ and $n^{(j)}$ denoting the number of points in cluster $i$ and class $j$. A high NMI indicates the clustering and class labels match well.

For computing the CR, each learned category is first associated with the "ground-truth" category that accounts for the largest number of samples in the learned category. Then the CR is computed as the ratio of the number of correctly classified samples to the size of the dataset. That is,

$$CR = \frac{\#\text{ correctly classified samples}}{\#\text{samples in the dataset}} \times 100\%. \tag{44}$$

Obviously, a higher CR indicates a more accurate clustering.

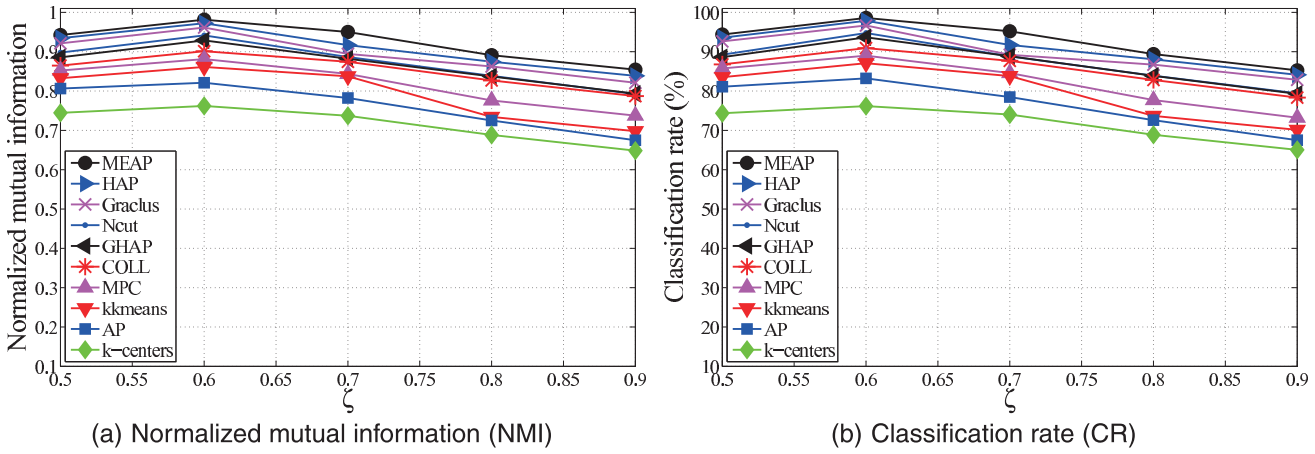



Fig. 7. The NMI and CR values as a function of $\zeta$ on JAFFE when using the actual number of categories, i.e., 10.

### 4.4 Results

#### 4.4.1 JAFFE Dataset

The JAFFE database [36] contains 213 images of seven facial expressions posed by 10 Japanese females. The seven facial expressions include six basic facial expressions, i.e., happiness, sadness, surprise, anger, disgust, fear, *plus* one neutral expression. There are three or four examples for each expression per person. Images of the same person in different facial expressions should be taken as in distinct subclasses [18], [25]. The goal is to cluster the 213 images into 10 groups according to identity.

We plot in Fig. 7 the clustering results when different $\zeta$ values are used in constructing the similarity metric. In this experiment, we run the algorithms in the different similarity metrics constructed with $\zeta$ ranging from 0.5 to 0.9, when using the actual number of categories, i.e., 10. In Figs. 7a and 7b, the NMI and CR values reported as a function of $\zeta$ have shown that on the JAFFE dataset, the most appropriate $\zeta$ is 0.6, and the proposed MEAP method has generated the best results. Hence, in the following comparison, we will use $\zeta = 0.6$ to construct the similarity metric on the JAFFE dataset.

We plot in Fig. 8 the complete results on the JAFFE dataset by MEAP when setting the preferences at the median of the similarities. In this setting, although the AP method results in a moderate number of clusters [1], yet it oversegments the images of 10 subjects into more clusters than the actual, with each cluster corresponding to a facial expression. When setting the preferences at the minimum of similarities, leading to a small number of clusters in AP, some images are misclassified by AP due to the presence of facial expression. However, our proposed MEAP can overcome the problem and almost perfectly group images of the same subject into one cluster by assigning exemplars

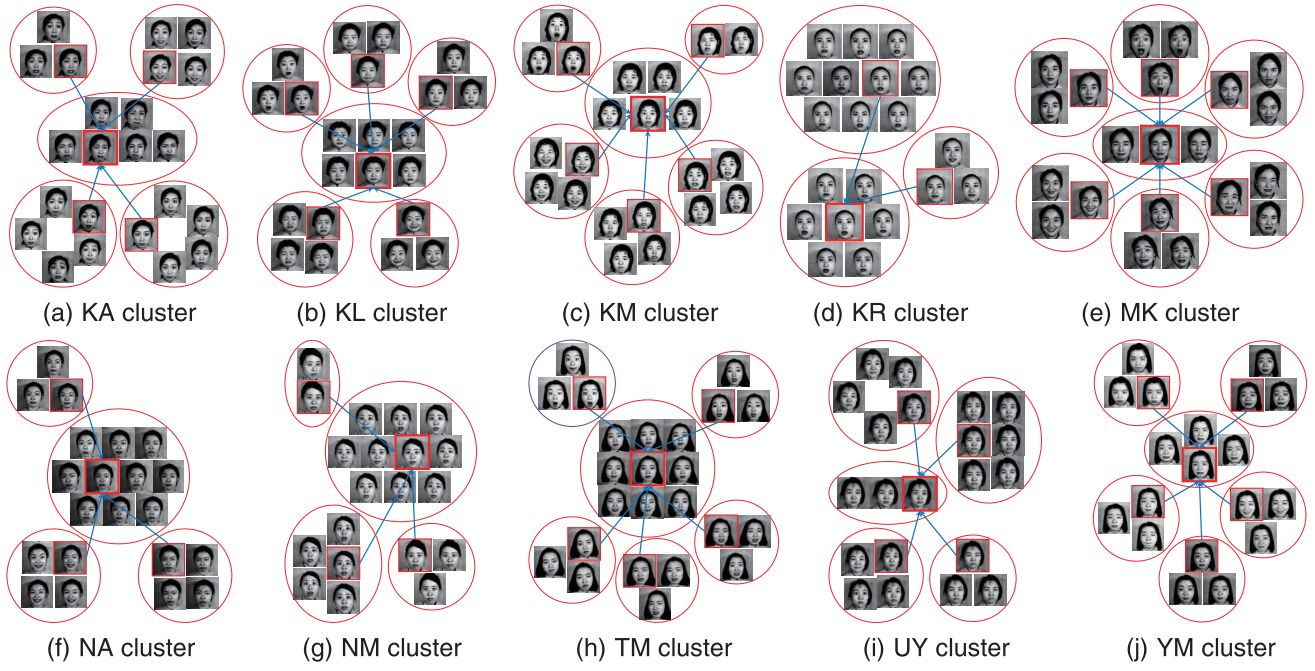



Fig. 8. Clusters learned by MEAP on the JAFFE dataset. The exemplars and superexemplars are bounded by thin and thick red lines, respectively. Each exemplar is connected to the corresponding superexemplar by the blue arrow. Among 213 images, only three images belonging to the YM category are misclassified to the TM category, as circled in purple, leading to a 98.6 percent CR.

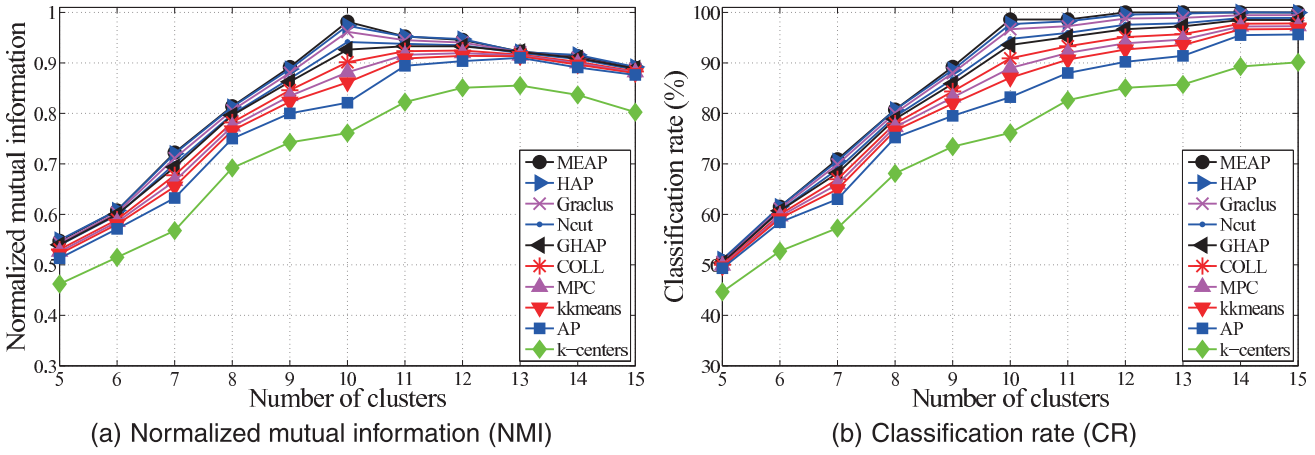



Fig. 9. The NMI and CR values as a function of the number of clusters learned on the JAFFE dataset.

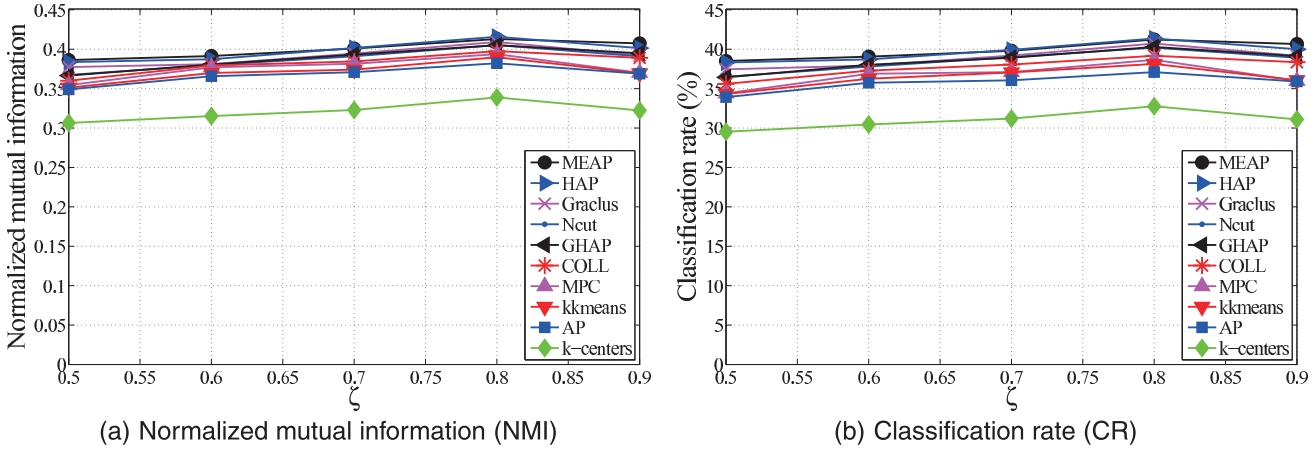



Fig. 10. The NMI and CR values as a function of $\zeta$ on Caltech101 when using the actual category number, i.e., 20.

of the same subject to the corresponding super-exemplar. Among 213 images, only three images belonging to the YM category are misclassified to the TM category, leading to a 98.6 percent CR. The reason is that the exemplar of these three images is more similar to the superexemplar of the TM category than YM. On the other hand, the AP algorithm misclassifies 36 images, resulting in an 83.1 percent CR, as will be demonstrated later.

We plot in Figs. 9a and 9b, respectively, the NMI and CR values as a function of $K$, the number of clusters. Notice that $K$ varies by adjusting the preference $s_{ii}$ over a range of values in AP and MEAP. And in other clustering methods, clusterings with different cluster numbers are also produced by preselecting in initialization. The best of 10,000 runs of $k$-centers with random initialization is plotted. Hereafter, this setting is used when reporting the NMI and CR values as a function of the cluster number. The figure shows that, in terms of NMI and CR, the proposed MEAP algorithm significantly outperforms its counterparts AP and $k$-centers consistently in the case of different cluster numbers, and generates comparable results when compared with other methods.

### 4.4.2 Caltech101 Dataset

The Caltech101 image dataset [37] contains 8,677 pictures of objects belonging to 101 categories. We use the same experimental setting as that in the work [9]. That is, 20 of the 101 classes representing a wide range of objects are selected, including (with numbers in parentheses denoting the number of images in each class) faces (435), leopards (200), motorbikes (798), binocular (33), brain (98), camera (50), car side (123), dollar bill (52), ferry (67), garfield (34), hedgehog (54), pagoda (47), rhino (59), snoopy (35), stapler (45), stop sign (64), water lilly (37), windsor chair (56), wrench (39), and yin yang (60). And only the first 100 images in the classes containing a very large number of images (e.g., faces, leopards, motorbikes, car side) are used, yielding a total dataset of 1,230 images. It should be noticed that, on this dataset, the presence of multiple subclasses within each category is not as obvious as on the JAFFE and

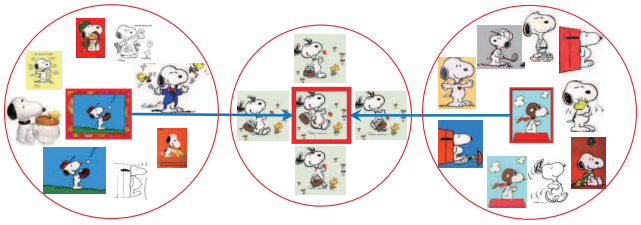
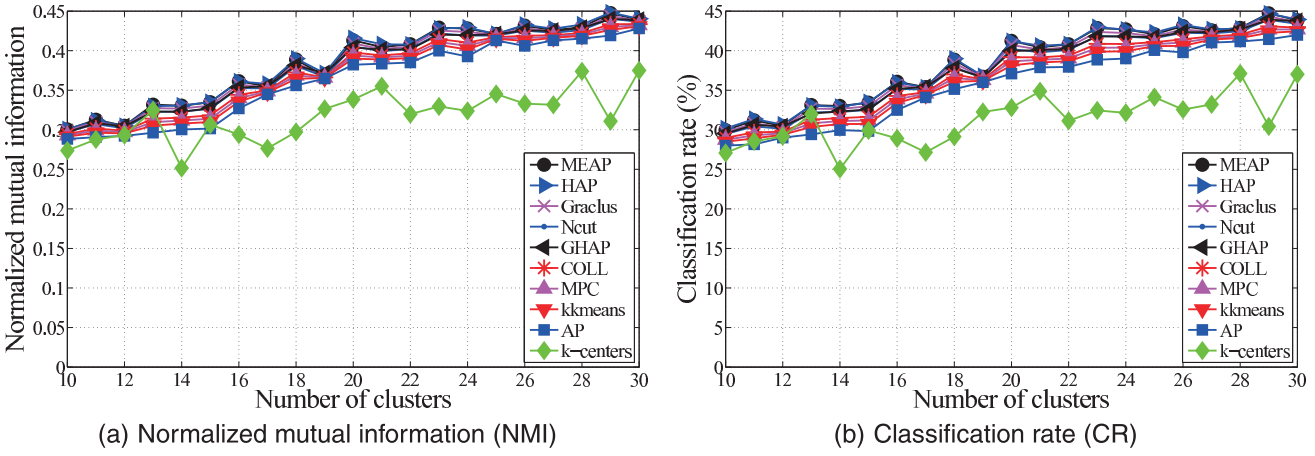
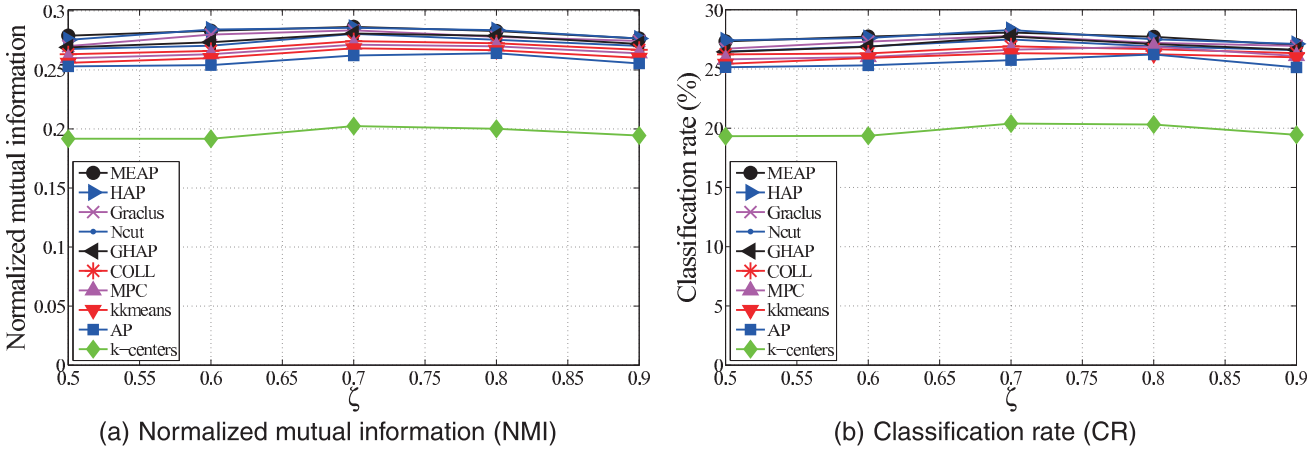


Fig. 11. Comparing AP and MEAP in the snoopy category from Caltech101. By setting the reference at the minimum value of the similarity matrix, AP has produced three categories associated with the snoopy category, as shown by the red circle. However, through assigning the exemplars to the most appropriate superexemplars (the blue arrow), MEAP can group these subcategories into one.




(a) Normalized mutual information (NMI) (b) Classification rate (CR)

Fig. 12. The NMI and CR values as a function of the number of clusters learned on the Caltech101 dataset.




(a) Normalized mutual information (NMI) (b) Classification rate (CR)

Fig. 13. The NMI and CR values as a function of $\zeta$ on SceneClass13 when using the actual category number, i.e., 13.

SceneClass13 datasets. However, as we will show, MEAP still outperforms AP in this case, which validates that MEAP is more flexible than AP.

Like on JAFFE, we first have to select the appropriate $\zeta$ for constructing the similarity metric on Caltech101. We plot in Fig. 10 the clustering results when different $\zeta$ values are used in constructing the similarity metric. In this experiment, we run the algorithms in the different similarity metrics constructed with $\zeta$ ranging from 0.5 to 0.9 when using the actual number of categories, i.e., 20. The NMI and CR results plotted, respectively, in Figs. 10a and 10b have shown that the most appropriate $\zeta$ value for the Caltech101 dataset is 0.8. Therefore, in later comparisons, we will use $\zeta = 0.8$ for constructing the similarity metric on Caltech101.

As reported in [9], the snoopy category was partitioned into a few subclasses. In the largest subclass associated with the ground-truth snoopy category as shown in [9, Fig. 5], some chair images were misclassified into the snoopy category. Fig. 11 shows the results by AP and MEAP when setting preferences at the minimum value of the similarity matrix. The AP algorithm produces three clusters associated with the snoopy category. However, MEAP can improve the results by avoiding oversegmentation through assigning exemplars to the most appropriate superexemplars.

We plot in Figs. 12a and 12b, respectively, the NMI and CR values as a function of $K$, the number of clusters. Although the existence of multiple subclasses is not so clear compared with that of JAFFE, the comparative results have shown that in terms of NMI and CR, the proposed MEAP still outperforms AP and $k$-centers consistently on this dataset.

### 4.4.3 SceneClass13 Dataset

The SceneClass13 image dataset [17] contains 3,759 images of 13 natural scene categories (with numbers in parentheses

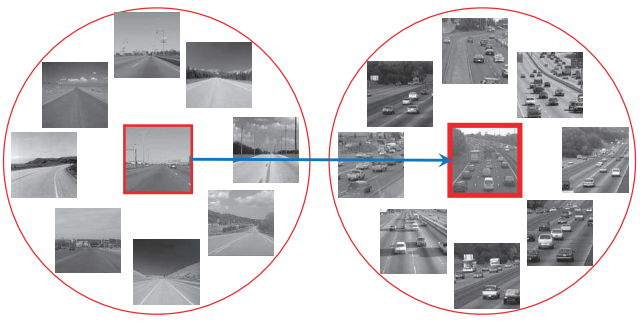
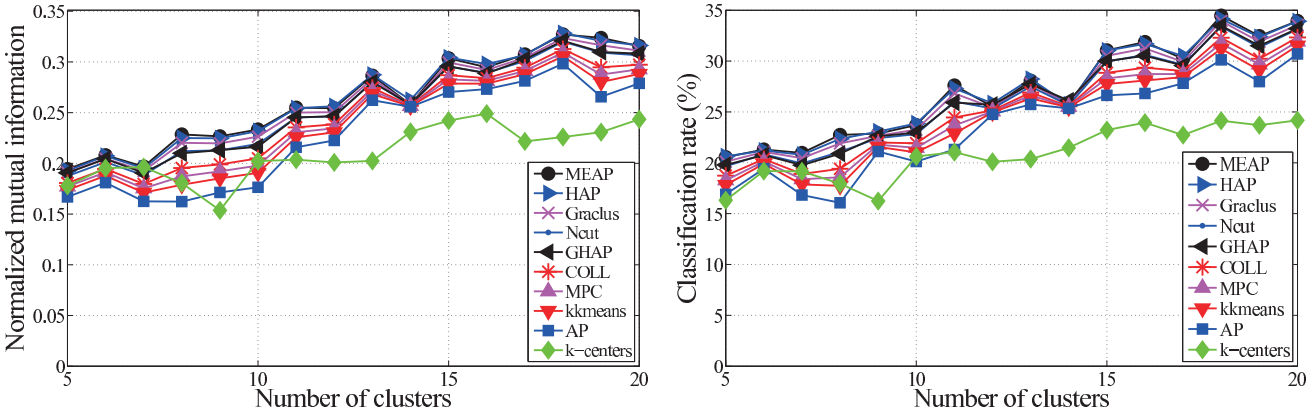


Fig. 14. Comparing AP and MEAP in the highway category from SceneClass13. By setting the reference at the minimum value of the similarity matrix, AP has produced two categories (with/without vehicles) associated with the highway category, as shown by the red circle. However, by assigning the exemplars to the most appropriate super-exemplars (the blue arrow), MEAP can group these two subcategories into one.




(a) Normalized mutual information (NMI) (b) Classification rate (CR)

Fig. 15. The NMI and CR values as a function of the number of clusters learned on the SceneClass13 dataset.

TABLE 1
The Average Values and Standard Deviations (in Parentheses) of NMI and CR, and the Average Computational Time in Seconds, over 100 Runs on the Three Image Datasets Using the Actual Cluster Number

| Methods | JAFFE | | | Caltech101 | | | SceneClass13 | | |
|---|---|---|---|---|---|---|---|---|---|
| | NMI | CR | Time | NMI | CR | Time | NMI | CR | Time |
| $k$-centers | 0.723(0.101) | 72.3%(0.101) | 0.951 | 0.328(0.099) | 31.7%(0.100) | 6.733 | 0.182(0.156) | 18.3%(0.149) | 20.320 |
| VSH | 0.834(0.103) | 84.6%(0.103) | 0.391 | 0.375(0.130) | 36.2%(0.127) | 6.386 | 0.201(0.148) | 20.6%(0.145) | 39.523 |
| AP | 0.821(0.000) | 83.1%(0.000) | 0.272 | 0.382(0.000) | 37.1%(0.000) | 1.249 | 0.262(0.000) | 25.7%(0.000) | 3.364 |
| GHAP | 0.926(0.000) | 93.5%(0.000) | 0.401 | 0.404(0.000) | 39.9%(0.000) | 2.317 | 0.280(0.000) | 27.8%(0.000) | 5.088 |
| HAP | 0.973(0.000) | 97.6%(0.000) | 0.399 | **0.413**(0.000) | **41.2**%(0.000) | 1.944 | **0.287**(0.000) | **28.1**%(0.000) | 5.239 |
| k$k$means | 0.860(0.100) | 87.3%(0.101) | 1.027 | 0.389(0.094) | 38.1%(0.091) | 7.098 | 0.268(0.128) | 26.3%(0.130) | 29.321 |
| MPC | 0.887(0.104) | 89.2%(0.107) | 2.743 | 0.394(0.102) | 38.7%(0.101) | 14.341 | 0.271(0.131) | 26.6%(0.131) | 41.236 |
| COLL | 0.901(0.008) | 91.1%(0.007) | 0.399 | 0.398(0.009) | 39.2%(0.008) | 2.403 | 0.274(0.030) | 26.9%(0.032) | 6.793 |
| Ncut | 0.942(0.013) | 94.8%(0.013) | 0.584 | 0.405(0.016) | 40.2%(0.015) | 3.290 | 0.281(0.023) | 27.5%(0.023) | 8.032 |
| Graclus | 0.962(0.012) | 96.7%(0.012) | 0.350 | 0.409(0.014) | 40.7%(0.014) | 2.112 | 0.283(0.020) | 27.8%(0.020) | 5.539 |
| MEAP | **0.981**(0.000) | **98.6**%(0.000) | 0.387 | **0.413**(0.000) | **41.2**%(0.000) | 1.892 | **0.287**(0.000) | **28.1**%(0.000) | 5.141 |

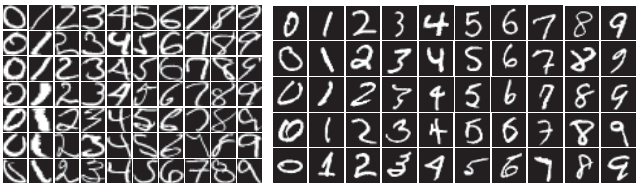
denoting the image number in each class): highway (260), inside of cities (308), tall buildings (356), streets (292), suburb residence (241), forest (328), coast (360), mountain (374), open country (410), bedroom (174), kitchen (151), living-room (289), and office (216). As discussed in [17], a scene category may contain multiple themes. For instance, the coast scene contains at least two typical themes (with and without "sunset/sunrise"), as shown in Fig. 1. The goal is to learn the scene categories consisting of multiple subclasses (themes).

Likewise, the NMI and CR values as a function of $\zeta$ plotted, respectively, in Figs. 13a and 13b show that the most appropriate $\zeta$ for the SceneClass13 dataset is 0.7 and it does not seriously affect the clustering results.

A significant improvement has been achieved on the dataset of SceneClass13. As illustrated in Fig. 14, MEAP can correctly model the highway category from other similar categories (e.g., the coast category as shown in Fig. 1) by integrating the two subclasses of the highway category (i.e., with/without "vehicles") into one cluster. On the contrary, AP separates the category into two clusters.

We plot in Figs. 15a and 15b, respectively, the NMI and CR values as a function of $K$, the number of clusters. Still, on this dataset, the proposed MEAP outperforms the $k$-centers and AP methods; meanwhile, it is comparable with other state-of-the-art clustering methods.

For the overall comparison on the performance of unsupervised image categorization, Table 1 lists the average values and standard deviations of NMI and CR, and the average computational time in seconds, over 100 runs on the three image datasets using the actual cluster number. Apart from the previously compared methods, another exemplar-based clustering method in operation research termed vertex substitution heuristic (VSH) [46] is performed and compared. From the table, the proposed MEAP algorithm significantly outperforms $k$-centers and AP, and is comparable with the state-of-the-art clustering methods. The advantage of MEAP is not only that it can model the category of more complex structure than AP without increasing the complexity of the generated model, but also that it can automatically estimate the number of clusters and the appropriate number of subclusters within each cluster when compared with existing multiexemplar clustering methods. The MPC method requires tuning many parameters to generate the desired number of clusters. On the other hand, MEAP inherits the advantages of AP in auto-initialization such that it does not need to tune any parameter [1]. Interestingly, the clustering performances of MEAP and HAP are very similar, with MEAP being slightly better than HAP in these unsupervised image categorization experiments. By observing the computational time, the time ratio of MEAP to AP is less than $7:3$. Although the proposed MEAP method is not the fastest, its time consumption is acceptable.



(a) USPS (b) MNIST

Fig. 16. Some samples of the USPS and MNIST datasets.

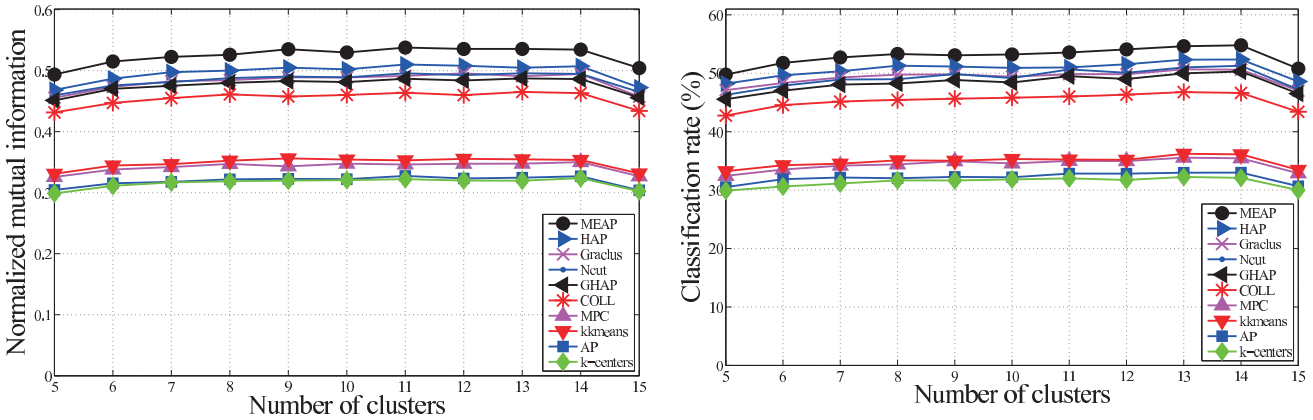
## 5 Handwritten Digits Clustering

This section reports the experimental results in the applications of handwritten digit clustering over two widely tested handwritten digit datasets, which are the USPS dataset [47] and the MNIST dataset [48]. As discussed in [19] and [20], the class representing a handwritten digit could be composed of several subclasses, each corresponding to a different handwriting style. The experimental results reported in this section coincide with this discussion and show that MEAP obtains more accurate clustering results than the compared methods.

The United States postal service (USPS) digit dataset contains 11,000 scaled handwritten digit images of size $16 \times 16$, with 1,100 images for each digit category [47]. The Modified National Institute of Standards and Technology (MNIST) dataset used in this paper contains 5,000 scaled handwritten digit images of size $28 \times 28$, with 500 images for each digit category. Some samples of the two datasets are shown in Figs. 16a and 16b, respectively.

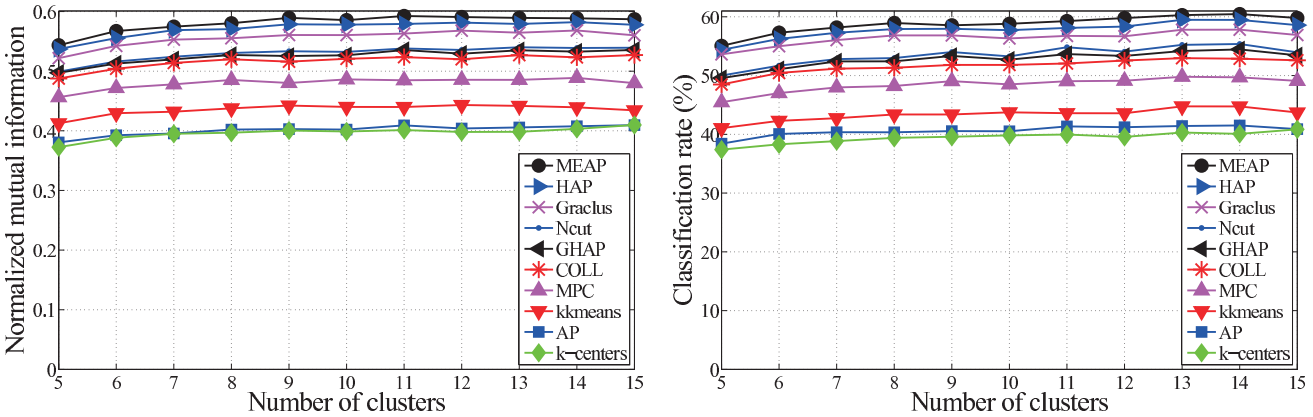
The experiments are performed in the gray scale pixels of digit images. The similarity matrix $[s_{ij}]_{N\times N}$ is computed as the negative euclidean distance, i.e., $s_{ij} = maxdist - \| \mathbf{x}_i - \mathbf{x}_j \|^2$ and the linkage matrix $[l_{ij}]_{N\times N}$ is set to $l_{ij} = s_{ij}/N$, where $maxdist$ denotes the $\max_{k,l\in\{1,\ldots,N\}} \| \mathbf{x}_k - \mathbf{x}_l \|^2$, $\mathbf{x}_i$ is the $i$th digit image, and $N$ is the size of the dataset. The same experimental settings are used as in the image categorization.

We plot in Figs. 17 and 18, respectively, the NMI and CR values as a function of the number of generated clusters. Table 2 lists the average values and standard deviations of NMI and CR, and the average computational time in seconds, over 100 runs on the two digit datasets using the actual cluster number. From the table, MEAP obtains 21.2 and 18.3 percent CR improvements, respectively, on USPS and MNIST over AP. This is a very significant improvement. When compared with the runner-up algorithm, namely, HAP, MEAP also outperforms HAP by obtaining 2.4 and 1.1 percent CR improvements, respectively, on USPS and MNIST. From the standard deviations of NMI and CR, we can see that the AP-like algorithms, namely, AP, GHAP, HAP, and MEAP, generate clustering results insensitive to initialization. By considering both the clustering accuracy and computational time, the comparative




(a) Normalized mutual information (NMI) (b) Classification rate (CR)

Fig. 17. The NMI and CR values as a function of the number of clusters learned on the USPS digit dataset.




(a) Normalized mutual information (NMI) (b) Classification rate (CR)

Fig. 18. The NMI and CR values as a function of the number of clusters learned on the MNIST digit dataset.

TABLE 2
The Average Values and Standard Deviations (in Parentheses) of NMI and CR, and the Average Computational Time in Seconds, over 100 Runs on the Two Digit Datasets Using the Actual Cluster Number

| Methods | USPS | | | MNIST | | |
|---|---|---|---|---|---|---|
| | NMI | CR | Time | NMI | CR | Time |
| $k$-centers | 0.317(0.163) | 31.4%(0.164) | 71.388 | 0.389(0.147) | 38.7%(0.150) | 32.794 |
| VSH | 0.318(0.155) | 31.9%(0.151) | 255.804 | 0.391(0.133) | 39.0%(0.137) | 85.381 |
| AP | 0.323(0.000) | 32.1%(0.000) | 10.493 | 0.402(0.000) | 40.5%(0.000) | 5.221 |
| GHAP | 0.481(0.000) | 48.4%(0.000) | 19.302 | 0.527(0.000) | 52.7%(0.000) | 10.377 |
| HAP | 0.502(0.000) | 50.9%(0.000) | 18.995 | 0.578(0.000) | 57.7%(0.000) | 10.015 |
| k$k$means | 0.354(0.154) | 35.2%(0.157) | 78.427 | 0.441(0.144) | 43.7%(0.143) | 34.040 |
| MPC | 0.347(0.179) | 34.6%(0.171) | 105.744 | 0.486(0.160) | 48.5%(0.159) | 50.983 |
| COLL | 0.461(0.081) | 45.8%(0.082) | 23.740 | 0.520(0.090) | 51.9%(0.091) | 10.779 |
| Ncut | 0.488(0.098) | 49.3%(0.099) | 31.231 | 0.532(0.093) | 53.2%(0.094) | 13.082 |
| Graclus | 0.490(0.094) | 49.5%(0.097) | 19.877 | 0.564(0.091) | 56.3%(0.090) | 9.499 |
| MEAP | **0.531**(0.000) | **53.3**%(0.000) | 18.910 | **0.584**(0.000) | **58.8**%(0.000) | 9.028 |

results have validated the effectiveness of the proposed MEAP method.

## 6 Conclusions

In this paper, we have extended the single-exemplar model to a multi-exemplar one and proposed a MEAP algorithm. In the multi-exemplar model, each data point is assigned to the most appropriate exemplar and each exemplar is assigned to the most appropriate superexemplar. Each cluster contains one superexemplar and an automatically determined number of exemplars assigned to that superexemplar. The objective is to maximize the sum of all similarities between data points and the corresponding exemplars *plus* the sum of all linkages between exemplars and the corresponding superexemplars. The max-sum belief propagation is utilized to solve the NP-hard optimization. By initializing all data points as exemplars and superexemplars and passing messages between data points and exemplars/superexemplars, between exemplars and superexemplars, MEAP produced clusters insensitive to initialization and converged to the neighborhood maximum.

The new MEAP algorithm has been found to be more effective than AP in the applications of multisubclasses clustering such as unsupervised image categorization and handwritten digit clustering. Image categorization experiments on three commonly tested datasets and handwritten digit clustering on two digit datasets have been conducted to compare MEAP with AP and $k$-centers, as well as other state-of-the-art clustering methods. The comparative results have confirmed the significant improvement made by our method.

## Acknowledgments

This work was supported by NSFC (61173084 and 61128009), NSFC-GuangDong (U0835005). The authors would like to thank Jianbo Shi for providing the Ncut code, Inderjit S. Dhillon for providing the Graclus code, Brendan J. Frey for providing the AP code, and Fei-Fei Li for providing the SceneClass13 and Caltech101 datasets.

## References

[1] B.J. Frey and D. Dueck, "Clustering by Passing Messages between Data Points," *Science,* vol. 315, pp. 972-976, http://www.psi.toronto.edu/index.php?q=affinity%20propagation, 2007.

[2] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger, "Factor Graphs and the Sum-Product Algorithm," *IEEE Trans. Information Theory,* vol. 47, no. 2, pp. 498-519, Feb. 2001.

[3] Y. Weiss and W.T. Freeman, "On the Optimality of Solutions of the Max-Product Belief-Propagation Algorithm in Arbitrary Graphs," *IEEE Trans. Information Theory,* vol. 47, no. 2, pp. 736-744, Feb. 2001.

[4] J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," *Proc. 15th Berkeley Symp. Math. Statistics and Probability,* vol. 1, pp. 281-297, 1967.

[5] A.K. Jain, "Data Clustering: 50 Years Beyond K-Means," *Pattern Recognition Letters,* vol. 31, pp. 651-666, 2010.

[6] D. Dueck, B.J. Frey, N. Jojic, V. Jojic, G. Giaever, A. Emili, G. Musso, and R. Hegele, "Constructing Treatment Portfolios Using Affinity Propagation," *Proc. 12th Ann. Int'l Conf. Research in Computational Molecular Biology,* pp. 360-371, 2008.

[7] T.-H. Huang, K.-Y. Cheng, and Y.-Y. Chuang, "A Collaborative Benchmark for Region of Interest Detection Algorithms," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 296-303, 2009.

[8] R. Verma and P. Wang, "On Detecting Subtle Pathology via Tissue Clustering of Multi-Parametric Data Using Affinity Propagation," *Proc. 11th IEEE Int'l Conf. Computer Vision,* pp. 1-8, 2007.

[9] D. Dueck and B.J. Frey, "Non-Metric Affinity Propagation for Unsupervised Image Categorization," *Proc. 11th IEEE Int'l Conf. Computer Vision,* pp. 1-8, 2007.

[10] Z.-Q. Zhao, J. Gao, H. Glotin, and X. Wu, "A Matrix Modular Neural Network Based on Task Decomposition with Subspace Division by Adaptive Affinity Propagation Clustering," *Applied Math. Modelling,* vol. 34, pp. 3884-3895, 2010.

[11] M.L. Sumedha and M. Weigt, "Unsupervised and Semi-Supervised Clustering by Message Passing: Soft-Constraint Affinity Propagation," *European Physical J. B,* vol. 66, pp. 125-135, 2008.

[12] D. Tarlow, R.S. Zemel, and B.J. Frey, "Flexible Priors for Exemplar-Based Clustering," *Proc. 24th Conf. Uncertainty in Artificial Intelligence,* pp. 537-545, 2008.

[13] X. Zhang, C. Furtlehner, and M. Sebag, "Data Streaming with Affinity Propagation," *Proc. European Conf. Machine Learning and Knowledge Discovery in Databases,* pp. 628-643, 2008.

[14] I.E. Givoni and B.J. Frey, "Semi-Supervised Affinity Propagation with Instance-Level Constraints," *Proc. Conf. Artificial Intelligence and Statistics,* pp. 161-168, 2009.

[15] J. Xiao, J. Wang, P. Tan, and L. Quan, "Joint Affinity Propagation for Multiple View Segmentation," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 1-7, 2007.

[16] I.E. Givoni, C. Chung, and B.J. Frey, "Hierarchical Affinity Propagation," *Proc. 24th Conf. Uncertainty in Artificial Intelligence,* pp. 238-246, 2011.

[17] L. Fei-Fei and P. Perona, "A Bayesian Hierarchical Model for Learning Natural Scene Categories," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 524-531, 2005.

[18] M. Zhu and A.M. Martinez, "Subclass Discriminant Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 8, pp. 1274-1286, Aug. 2006.

[19] H.I. Avi-Itzhak, J.A.V. Mieghem, and L. Rub, "Multiple Subclass Pattern Recognition: A Maximin Correlation Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 4, pp. 418-431, Apr. 1995.

[20] A.F.R. Rahman and M. Fairhurst, "Multi-Prototype Classification: Improved Modelling of the Variability of Handwritten Data Using Statistical Clustering Algorithms," *Electronics Letters,* vol. 33, pp. 1208-1210, 1997.

[21] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation,* vol. 10, pp. 1299-1319, 1998.

[22] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[23] C.-D. Wang, J.-H. Lai, and J.-Y. Zhu, "A Conscience On-Line Learning Approach for Kernel-Based Clustering," *Proc. 10th Int'l Conf. Data Mining,* pp. 531-540, 2010.

[24] I.S. Dhillon, Y. Guan, and B. Kulis, "Weighted Graph Cuts Without Eigenvectors: A Multilevel Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 11, pp. 1944-1957, Nov. 2007.

[25] S.K. Zhou and R. Chellappa, "Multiple-Exemplar Discriminant Analysis for Face Recognition," *Proc. 17th Int'l Conf. Pattern Recognition,* pp. 191-194, 2004.

[26] J. Reisinger and R.J. Mooney, "Multi-Prototype Vector-Space Models of Word Meaning," *Proc. Ann. Conf. North Am. Chapter of the Assoc. for Computational Linguistics,* pp. 109-117, 2010.

[27] Q. Zhu, Y. Cai, and L. Liu, "A Multiple Hyper-Ellipsoidal Subclass Model for an Evolutionary Classifier," *Pattern Recognition,* vol. 34, pp. 547-560, 2001.

[28] F. Aiolli and A. Sperduti, "Multiclass Classification with Multi-Prototype Support Vector Machines," *J. Machine Learning Research,* vol. 6, pp. 817-850, 2005.

[29] S. Guha, R. Rastogi, and K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," *Information Systems,* vol. 26, no. 1, pp. 35-58, 2001.

[30] M. Liu, X. Jiang, and A.C. Kot, "A Multi-Prototype Clustering Algorithm," *Pattern Recognition,* vol. 42, pp. 689-698, 2009.

[31] T. Luo, C. Zhong, H. Li, and X. Sun, "A Multi-Prototype Clustering Algorithm Based on Minimum Spanning Tree," *Proc. Seventh Int'l Conf. Fuzzy Systems and Knowledge Discovery,* pp. 1602-1607, 2010.

[32] D. Dueck, "Affinity Propagation: Clustering Data by Passing Messages," PhD dissertation, Univ. of Toronto, 2009.

[33] I.E. Givoni and B.J. Frey, "A Binary Variable Model for Affinity Propagation," *Neural Computation,* vol. 21, no. 6, pp. 1589-1600, June 2009.

[34] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis.* Cambridge Univ. Press, 2004.

[35] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum Margin Clustering," *Proc. Conf. Neural Information Processing Systems,* 2004.

[36] M.J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding Facial Expressions with Gabor Wavelets," *Proc. Third IEEE Int'l Conf. Automatic Face and Gesture Recognition,* pp. 200-205, 1998.

[37] L. Fei-Fei, R. Fergus, and P. Perona, "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories," *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshop,* pp. 178-188, 2004.

[38] D. You, O.C. Hamsici, and A.M. Martinez, "Kernel Optimization in Discriminant Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 33, no. 3, pp. 631-638, Mar. 2011.

[39] K. Grauman and T. Darrell, "Unsupervised Learning of Categories from Sets of Partially Matching Image Features," *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshop,* pp. 2596-2603, 2006.

[40] K. Mikolajczyk, B. Leibe, and B. Schiele, "Multiple Object Class Detection with a Generative Model," *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshop,* pp. 26-36, 2006.

[41] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[42] A. Strehl and J. Ghosh, "Cluster Ensembles—A Knowledge Reuse Framework for Combining Multiple Partitions," *J. Machine Learning Research,* vol. 3, pp. 583-617, 2002.

[43] A. Strehl, J. Ghosh, and R.J. Mooney, "Impact of Similarity Measures on Web-Page Clustering," *Proc. AAAI Workshop AI for Web Search,* pp. 58-64, 2000.

[44] L. Hubert and P. Arabie, "Comparing Partitions," *J. Classification,* vol. 2, pp. 193-218, 1985.

[45] M. Meil, "Comparing Clusterings—An Axiomatic View," *Proc. 22nd Int'l Conf. Machine Learning,* pp. 577-584, 2005.

[46] P. Hansen and N. Mladenović, "Variable Neighborhood Search for the p-Median," *Location Science,* vol. 5, no. 4, pp. 207-226, Dec. 1997.

[47] J.J. Hull, "A Database for Handwritten Text Recognition Research," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 16, no. 5, pp. 550-554, May 1994.

[48] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE,* vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

**Chang-Dong Wang** received the BS degree in applied mathematics and the MSc degree in computer science from Sun Yat-sen University, China, in 2008 and 2010, respectively. Since September 2010, he has been working toward the PhD degree at Sun Yat-sen University. He was a visiting student at the University of Illinois at Chicago from January 2012 to January 2013. His current research interests include machine learning and pattern recognition, especially focusing on data clustering and its applications. He has published several scientific papers in international journals and conferences such as the *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Systems, Man, and Cybernetics-C*, KAIS, Neurocomputing, and ICDM. His ICDM '10 paper won the Honorable Mention for Best Research Paper Awards. He won the Student Travel Award from ICDM '10 and ICDM '11, respectively. He is a student member of the IEEE.

**Jian-Huang Lai** received the MSc degree in applied mathematics and the PhD degree in mathematics from Yat-sen University, China, in 1989 and 1999, respectively. He joined Sun Yat-sen University in 1989 as an assistant professor, where he is currently a professor with the Department of Automation of School of Information Science and Technology and vice dean of the School of Information Science and Technology. His current research interests include the areas of digital image processing, pattern recognition, multimedia communication, wavelet, and its applications. He has published more than 100 scientific papers in international journals and conferences on image processing and pattern recognition, for example, the *IEEE Transactions on Neural Networks*, *IEEE TIP*, *IEEE Transactions on Systems, Man, and Cybernetics-B*, *Pattern Recognition*, ICCV, CVPR, and ICDM. He serves as a standing member of the Image and Graphics Association of China and also serves as a standing director of the Image and Graphics Association of Guangdong. He is a senior member of the IEEE.

**Ching Y. Suen** received the MSc degree in engineering from the University of Hong Kong and the PhD degree from the University of British Columbia, Vancouver, BC, Canada. He is the director of Centre for Pattern Recognition and Machine Intelligence, Concordia University, Montreal, QC, Canada, and the Concordia Chair on Artificial Intelligence and Pattern Recognition. He has guided/hosted 80 visiting scientists and professors, and has supervised 82 doctoral and master's graduates. Currently, he is the editor-in-chief of the journal of *Pattern Recognition* and an advisory or associate editor of four other journals. He has founded and organized numerous international conferences on pattern recognition, handwriting recognition, and document analysis. He has also founded the IAPR ICDAR Awards. He has served numerous professional societies as president, vice-president, governor, and director. He has given 180 invited talks at various industries and academic institutions around the world, and has been the principal investigator or consultant of 30 industrial projects. His publications include four conference proceedings, 12 books, and more than 480 papers, of which many have been widely cited. He is a fellow of the IAPR, and the Academy of Sciences of the Royal Society of Canada. He is a life fellow of the IEEE.

**Jun-Yong Zhu** received the BS and MS degrees from the School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou, P.R. China, in 2008 and 2010, respectively. He is currently working toward the PhD degree in the Department of Mathematics, Sun Yat-sen University. His current research interests include machine learning, transfer learning using auxiliary data, pattern recognition such as heterogeneous face recognition. He is a student member of the IEEE.