2014 22nd International Conference on Pattern Recognition

Multi-View Nonnegative Matrix Factorization for Clothing Image Characterization

Wei-Yi Chang
Research Center for IT Innovation
Academia Sinica, Taipei, Taiwan
Email: bill20312@gmail.com

Chia-Po Wei
Research Center for IT Innovation
Academia Sinica, Taipei, Taiwan
Email: cpwei@citi.sinica.edu.tw

Yu-Chiang Frank Wang
Research Center for IT Innovation
Academia Sinica, Taipei, Taiwan
Email: ycwang@citi.sinica.edu.tw

***Abstract*—Due to the ambiguity in describing and discriminating between clothing images of different styles, it has been a challenging task to solve clothing image characterization problems. Based on the use of multiple types of visual features, we propose a novel multi-view nonnegative matrix factorization (NMF) algorithm for solving the above task. Our multi-view NMF not only observes image representations for describing clothing images in terms of visual appearances, an optimal combination of such features for each clothing image style would also be learned, while the separation between different image styles can be preserved. To verify the effectiveness of our method, we conduct experiments on two image datasets, and we confirm that our method produces satisfactory performance in terms of both clustering and categorization.**

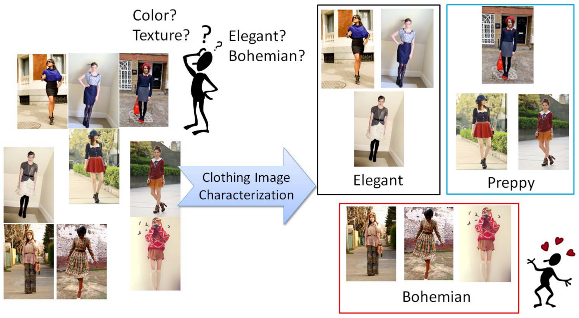


Fig. 1. Illustration of clothing image characterization for style categorization.

I. Introduction

Style discovery and recommendation for clothing images have been attracting the attention from the researchers in the fields of computer vision and pattern recognition. As illustrated in Figure 1, proper representation and categorization of such image data would allow users to select the clothing items of interest, which will be beneficial to a variety of industries such as fashion and e-commerce. Unfortunately, style (or fashion) of clothing images is an implicit concept, and it is typically very difficult to give proper definitions to each clothing style in terms of its visual appearances. In other words, how to sufficiently describe clothing images for each style using its color, texture, etc. visual features is still a very challenging task. In this paper, we particularly advance clustering-based algorithms which group the clothing images into different styles via multiple visual features, aiming at utilizing the observed image groups for characterizing the clothing styles.

As noted above, different types of visual features are often applied to represent clothing images, with the goal of separating them into different style categories for recommendation purposes. Among the visual features, color, texture, and shape descriptors are the most popular ones in describing clothing images [1]. Since the use of a single type of visual features is typically not able to provide sufficient or satisfactory representation capabilities, it is desirable to combine the information extracted from multiple features for improved performance. However, features in different visual domains are not necessarily complementary to each other. Moreover, the integration of multiple features does not guarantee the discrimination between different image styles. Thus, how to properly combine such features has been an ongoing task to be tackled for the researchers in related fields.

Existing clustering-based data representation algorithms with multiple feature integration can typically be divided into two groups: early and late fusion [2]. The former essentially considers all features of interest simultaneously when performing clustering [3], [4]. For example, Tzortis *et al.* [3] considered each feature type as a particular view, and proposed *weighted multiview convex mixture models (CMM)* to derive proper weights for each view when performing clustering. For late fusion approaches, one typically performs clustering using each individual feature/view first, and the corresponding results will be integrated for determining the final clustering output [5], [6], [7]. For example, Long *et al.* [5] applied mapping functions for associating and clustering data in terms of different features, and Cai *et al.* [6] proposed a multi-modal spectral clustering algorithm to integrate heterogenous image feature. To handle large-scale data with different features, Cai *et al.* [7] proposed a multi-view K-means clustering algorithm for late feature fusion with robustness to the presence of outlier data. Compared with the methods of early fusion, one of the advantages of late fusion is its distributed capability of being processed in parallel.

Aiming at observing a set of basis functions from the input data, nonnegative matrix factorization (NMF) [8], [9] was originally proposed for the purpose of dimension reduction. With the nonnegative constraints imposed on both the derived bases and coefficients, the input data will be represented as a nonnegative linear combination of the associated basis vectors. NMF has been applied for data analysis applications such as microarray data analysis [10], image analysis [11], image classification [12], and information retrieval/recommendation [13]. As discussed in [14], [15], NMF can be viewed as an effective clustering technique for data which exhibits such nonnegative

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.228

IEEE computer society

properties (e.g., images). In this paper, we propose a novel *multi-view NMF* algorithm for representing and clustering clothing images. Given the style labels of input clothing images, our proposed NMF not only utilizes multiple types of visual descriptors in characterizing such data, additional *discriminating* capabilities are also introduced into our NMF formulation. As a result, we are able to identify the contributions of each visual descriptor for different clothing image styles, while the separation between the styles of interest can be preserved.

The remaining of this paper is organized as follows. Section II briefly reviews NMF and its use for data clustering. Our proposed multi-view NMF will be detailed in Section III, followed by experiments on two image datasets presented in Section IV. Finally, Section V concludes this paper.

II. Related Work

A. Overview of NMF

With N data instances observed in an M-dimensional space, we construct the data matrix $\boldsymbol{X}$ of size $M \times N$. The goal of Nonnegative Matrix Factorization (NMF) is to decompose such a data matrix into two nonnegative matrices $\boldsymbol{U}$ and $\boldsymbol{V}$, so that the input $\boldsymbol{X}$ can be approximated by $\boldsymbol{U}\boldsymbol{V}^\top$. The former matrix of $\boldsymbol{U}$ is viewed as a basis matrix, while the latter matrix $\boldsymbol{V}$ represents the associated coefficients. The dimensions of matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are $M \times L$ and $N \times L$, respectively. We note that L indicates the number of basis vectors (and thus the number of coefficients for each data instances).

More precisely, NMF solves the following problem:

$$\min_{\boldsymbol{U},\boldsymbol{V}} \|\boldsymbol{X}-\boldsymbol{U}\boldsymbol{V}^\top\|_F^2, \quad s.t \quad \boldsymbol{U}\geq 0, \boldsymbol{V}\geq 0, \tag{1}$$

where $\|\cdot\|_F$ represents the Frobenius norm , and $\boldsymbol{U}\geq 0, \boldsymbol{V}\geq 0$ indicate that all elements in $\boldsymbol{U}$ and $\boldsymbol{V}$ are nonnegative. To solve the above optimization problem, Lee *et al.* [8] applied the technique of multiplicative updates, which derive the basis and coefficients matrices at each iteration as follows:

$$\boldsymbol{U}_{i,j} \leftarrow \boldsymbol{U}_{i,j}\frac{(\boldsymbol{X}\boldsymbol{V})_{i,j}}{(\boldsymbol{U}\boldsymbol{V}^\top\boldsymbol{V})_{i,j}}, \quad \boldsymbol{V}_{i,j} \leftarrow \boldsymbol{V}_{i,j}\frac{(\boldsymbol{X}^\top\boldsymbol{U})_{i,j}}{(\boldsymbol{V}\boldsymbol{U}^\top\boldsymbol{U})_{i,j}}, \tag{2}$$

where i and j are the indexes of the elements in the corresponding matrix. The iteration terminates when the solutions converge, or the maximum number of iteration is reached.

B. NMF for Data Clustering

Due to the nonnegative constraints imposed on $\boldsymbol{U}$ and $\boldsymbol{V}$, NMF views input instances as additive combinations of the derived basis vectors. As a result, it is interpretable for one to utilize the resulting NMF coefficients $\boldsymbol{V}$ as the resulting features for clustering, etc. tasks. Recently, researchers further extend NMF for determining improved representation for achieving the goal of clustering. For example, Cai *et al.* [16] proposed a graph regularized NMF to model manifold structures of input data. By advancing the nearest neighbor graph structures, a more discriminative representation can thus be obtained. On the other hand, Kong *et al.* [15] extended NMF by applying the $\mathrm{L}_{2,1}$ norm as the loss function. The use of such functions provides additional robustness in handling outlier or noisy data. To further take the data label information into consideration, Liu *et al.* [17] advocated a semi-supervised matrix decomposition technique which incorporates the label information as additional constraints in NMF. By forcing the data instances of the same image category closer to each other, their derived representations $\boldsymbol{V}$ were shown to be preferable for image clustering.

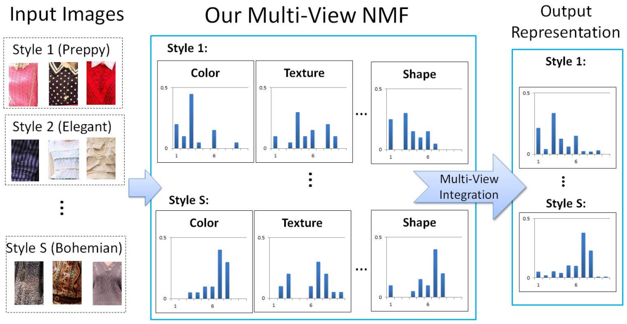


Fig. 2. Flowchart of our multi-view NMF for image characterization.

In addition to the above extensions, researchers also consider to perform NMF when taking multiple types of features into consideration [18], [19]. For example, Greene *et al.* [18] presented a simple yet effective algorithm for combining multi-view (multi-feature) data. They utilized the clustering results from each view and constructed an intermediate matrix representation, which allows the factorization using nonnegative components. However, the clustering results derived from each view were treated equally important in [18]. Later Liu *et al.* [19] proposed a NMF framework which pushes clustering results of each view into a common consensus. While not requiring the weights of each feature type to be the same, they assumed that such features would share similar clustering results, which might still limit the clustering performance.

We note that the above consensus-based strategies might not be applicable for style discovery/clustering of clothing images. For some clothes in one style, they might be visually similar to each other in terms of color, while those in another style might be more relevant in terms of texture information. This is the reason why we propose a novel multi-view NMF in this work, which focuses on exploiting the visual information within and across clothing image styles, while the introduced discriminative constraint would preserve the discrimination between different image styles.

III. Our Proposed Method

As illustrated in Figure 2, our multi-view NMF deals with input data in terms of different features (i.e., views), with the goal of determining an optimized integration of different coefficient matrices which preserves the separation between different clothing image styles. In Section III-A, we present our proposed multi-view NMF algorithm. The initialization of our algorithm will be discussed in Section III-B, while the optimization of our multi-view NMF will be detailed in Section III-C.

A. Multi-View Nonnegative Matrix Factorization

Consider that the images are collected from S classes (styles) and in terms of F different features (views). For each class $s \in \{1, 2, \ldots, S\}$ and view $f \in \{1, 2, \ldots, F\}$, we have the data matrix $\boldsymbol{X}_s^f$ of size $M_f \times N_s$, where M_f is the dimension for the image data in view f, and N_s is the number of images in class s.

We propose a multi-view NMF algorithm, which solves the following optimization problem:

$$\begin{aligned} &\min_{\boldsymbol{U},\boldsymbol{V},\alpha} \sum_{s=1}^{S} \{ \sum_{f=1}^{F} \|\boldsymbol{X}_s^f - \boldsymbol{U}_s^f (\boldsymbol{V}_s^f)^\top \|_F^2 \} + \eta \sum_{j \neq s}^{S} \|\boldsymbol{V}_j^* (\boldsymbol{V}_s^*)^\top \|_F^2, \\ &\text{s.t} \quad \boldsymbol{U}_s^f \geq 0, \boldsymbol{V}_s^f \geq 0, \boldsymbol{V}_s^* = \sum_{f=1}^{F} \alpha_s^f \boldsymbol{V}_s^f, \qquad (3) \\ &\qquad \sum_{f=1}^{F} \alpha_s^f = 1, \alpha_s^f \geq 0. \end{aligned}$$

It can be seen that, the first term in (3) aims at deriving nonnegative basis matrices $\boldsymbol{U}_s^f \in \mathbb{R}^{M_f \times L}$ and coefficient matrices $\boldsymbol{V}_s^f \in \mathbb{R}^{N_s \times L}$ for images in each class at a particular view. On the other hand, the regularization term (i.e., the second term) in (3) associates the information across different classes for data separation. Inspired by [20], this regularization term advocates the *structural incoherence* between the $\boldsymbol{V}^*$ of different classes across all views, and minimizing such incoherence would introduce additional *discriminating* capability into our multi-view NMF algorithm.

As shown in (3), for class s, its coefficient matrix $\boldsymbol{V}_s^*$ is a weighted combination of $\boldsymbol{V}_s^f$ observed in different views, and thus can be viewed as a final feature representation for that class. We note that, the weights α_s^f for each $\boldsymbol{V}_s^*$ are learned during the optimization process. Due to the advocate of structural incoherence in our multi-view NMF formulation, optimal α_s^f will be derived not only for image representation but also for class separation purposes. In other words, image features which are strongly relevant to the corresponding class will be assigned larger weights, but the difference in the final coefficient matrices will still be preserved by enforcing the above structural incoherence. We note that η in (3) balances the capabilities of image representation and structural incoherence for our multi-view NMF.

B. Initialization of $\boldsymbol{U}$ and $\boldsymbol{V}$

Since the weighted coefficient matrix $\boldsymbol{V}_s^*$ in (3) can be considered as the feature representation for class s, the basis matrix $\boldsymbol{U}_s^f$ and the corresponding coefficient matrix $\boldsymbol{V}_s^f$ in each view need to be properly initialized before solving (3). More precisely, the basis vectors of $\boldsymbol{U}_s^f$ at different views need to be aligned, so that the corresponding coefficient matrices $\boldsymbol{V}_s^f$ across different views can be directly combined for producing the final $\boldsymbol{V}_s^*$.

Now we discuss how this initialization is performed prior to the optimization of our multi-view NMF. For a particular view $f \in \{1, 2, \ldots, F\}$, we have data matrix $\boldsymbol{X}^f$ from all S classes, i.e., $\boldsymbol{X}^f = [\boldsymbol{X}_1^f, \boldsymbol{X}_2^f, \ldots, \boldsymbol{X}_S^f]$ of size $M_f \times N$, where $N = \sum_{s=1}^{S} N_s$. We concatenate such matrices across different views, and thus have the resulting data matrix $\boldsymbol{X}_{con} = [\boldsymbol{X}^1; \boldsymbol{X}^2; \ \ldots; \boldsymbol{X}^F]$ of size $M \times N$, where $M = \sum_{f=1}^{F} M_f$. We apply standard NMF to factorize $\boldsymbol{X}_{con}$ into $\boldsymbol{U}_{con}\boldsymbol{V}_{con}^\top$, where $\boldsymbol{U}_{con} \in \mathbb{R}^{M \times L}$, and $\boldsymbol{V}_{con} \in \mathbb{R}^{N \times L}$. It can be seen that, $\boldsymbol{U}_{con}$ represents the standard NMF basis matrix given the concatenated input data, and thus $\boldsymbol{U}_{con} = [\boldsymbol{U}_{con}^1; \ \ldots; \boldsymbol{U}_{con}^F]$.

Algorithm 1 Initialization of $\boldsymbol{U}$ and $\boldsymbol{V}$

```
Input: Data matrices from $F$ views $\{\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots, \boldsymbol{X}^F\}$ and the associated class labels
Output: Basis matrices $\{\tilde{\boldsymbol{U}}^1, \tilde{\boldsymbol{U}}^2, \ \ldots, \tilde{\boldsymbol{U}}^F\}$ and coefficient matrices $\{\tilde{\boldsymbol{V}}_1^1, \ \ldots, \tilde{\boldsymbol{V}}_S^F\}$
Concatenate data matrices: $[\boldsymbol{X}^1; \boldsymbol{X}^2; \ \ldots; \boldsymbol{X}^F] = \boldsymbol{X}_{con}$
Use NMF to factorize $\boldsymbol{X}_{con}$ as $\boldsymbol{U}_{con}\boldsymbol{V}_{con}^\top$
for $f = 1$ to $F$ do
    $(\tilde{\boldsymbol{U}}^f, \tilde{\boldsymbol{V}}^f) =$ NMF($\boldsymbol{X}^f$) with $\boldsymbol{U}_{con}^f$ and $\boldsymbol{V}_{con}$ as initial values
end for
return $\tilde{\boldsymbol{U}}^f$ and $\tilde{\boldsymbol{V}}^f = [\tilde{\boldsymbol{V}}_1^f; \ \ldots; \tilde{\boldsymbol{V}}_S^f]$ for each view $f$
```

Next, at each individual view f across all S classes, we refine the NMF of $\boldsymbol{X}^f$ using the above basis matrix $\boldsymbol{U}_{con}^f$ and coefficients $\boldsymbol{V}_{con}$ as initialization. The outputs of this NMF are $\tilde{\boldsymbol{U}}^f$ and $\tilde{\boldsymbol{V}}^f$, which will be viewed as final aligned basis and coefficient matrices at that view, respectively. As a result, the sub-matrix $\tilde{\boldsymbol{V}}_s^f$ (of size $N_s \times L$) of $\tilde{\boldsymbol{V}}^f$ can be viewed as the coefficient matrix for view f and class s derived from standard NMF, while such matrices of different classes share the same (aligned) basis matrix $\tilde{\boldsymbol{U}}^f$. In other words, we take $\tilde{\boldsymbol{U}}^f$ and $\tilde{\boldsymbol{V}}^f = [\tilde{\boldsymbol{V}}_1^f; \ \ldots; \tilde{\boldsymbol{V}}_S^f]$ for initializing the optimization of our multi-view NMF.

C. Optimization

To solve the optimization problem of (3), we apply the techniques of multiplicative updates [8]. In other words, in order to search for the optimal $\boldsymbol{U}_s^f$, $\boldsymbol{V}_s^f$, and α_s^f, we iterate between the following steps until convergence:

1) Updates of $\boldsymbol{U}_s^f$ and $\boldsymbol{V}_s^f$: With α_s^f fixed in (3) and by advancing the property of $\|\boldsymbol{V}_j^*(\boldsymbol{V}_s^*)^\top\|_F^2 \leq \|\boldsymbol{V}_j^*\|_F^2 \|\boldsymbol{V}_s^*\|_F^2$, we convert the original multi-view NMF objective function into the following relaxed version for each class:

$$\begin{aligned} &\min_{\boldsymbol{U},\boldsymbol{V}} \sum_{f=1}^{F} \|\boldsymbol{X}_s^f - \boldsymbol{U}_s^f (\boldsymbol{V}_s^f)^\top \|_F^2 + \eta' \|\boldsymbol{V}_s^*\|_F^2, \\ &\text{s.t} \quad \boldsymbol{U}_s^f \geq 0, \boldsymbol{V}_s^f \geq 0 \end{aligned} \qquad (4)$$

where $\eta' = \eta \sum_{j \neq s} \|\boldsymbol{V}_j^*\|_F^2$ is a constant when deriving $\boldsymbol{U}_s^f$ and $\boldsymbol{V}_s^f$ for class s. As a result, for class s in a specific view f, we can further simplify the above formulation into the following problem:

$$\begin{aligned} &\min_{\boldsymbol{U}_s^f, \boldsymbol{V}_s^f} \|\boldsymbol{X}_s^f - \boldsymbol{U}_s^f (\boldsymbol{V}_s^f)^\top \|_F^2 + \eta' \|\boldsymbol{B} + \alpha_s^f \boldsymbol{V}_s^f\|_F^2, \\ &\text{s.t} \quad \boldsymbol{U}_s^f \geq 0, \boldsymbol{V}_s^f \geq 0 \end{aligned} \qquad (5)$$

where $\boldsymbol{B} = \sum_{j \neq s} \alpha_j^f \boldsymbol{V}_j^f$. For simplicity, we drop the notations s and f in (5) and use the following formulation:

$$\begin{aligned} &\min_{\boldsymbol{U},\boldsymbol{V}} \|\boldsymbol{X} - \boldsymbol{U}\boldsymbol{V}^\top\|_F^2 + \eta' \|\boldsymbol{B} + \alpha \boldsymbol{V}\|_F^2 \\ &\text{s.t} \quad \boldsymbol{U} \geq 0, \boldsymbol{V} \geq 0. \end{aligned} \qquad (6)$$

Let Φ and Ψ be the Lagrange multipliers for the constraints $\boldsymbol{U} \geq 0, \boldsymbol{V} \geq 0$, we have the Lagrange function L as:

$$\begin{aligned} L = & Tr(\boldsymbol{X}\boldsymbol{X}^\top) - 2Tr(\boldsymbol{X}\boldsymbol{V}\boldsymbol{U}^\top) + Tr(\boldsymbol{U}\boldsymbol{V}^\top\boldsymbol{V}\boldsymbol{U}^\top) + \\ & \eta' Tr(\boldsymbol{B}\boldsymbol{B}^\top) + 2\eta'\alpha Tr(\boldsymbol{B}\boldsymbol{V}^\top) + \eta'\alpha^2 Tr(\boldsymbol{V}\boldsymbol{V}^\top) + \\ & Tr(\Phi\boldsymbol{U}) + Tr(\Psi\boldsymbol{V}). \end{aligned} \tag{7}$$

We take the partial derivatives of L with respect to $\boldsymbol{U}$ and $\boldsymbol{V}$:

$$\frac{\partial L}{\partial \boldsymbol{U}} = -2\boldsymbol{X}\boldsymbol{V} + 2\boldsymbol{U}\boldsymbol{V}^\top\boldsymbol{V} + \Phi^\top, \tag{8}$$

$$\frac{\partial L}{\partial \boldsymbol{V}} = -2\boldsymbol{X}^\top\boldsymbol{U} + 2\boldsymbol{V}\boldsymbol{U}^\top\boldsymbol{U} + 2\eta'\alpha(\boldsymbol{B} + \alpha\boldsymbol{V}) + \Psi^\top. \tag{9}$$

With Karush-Kuhn-Tucker (KKT) conditions $\Phi_{i,j}\boldsymbol{U}_{i,j} = 0$ and $\Psi_{i,j}\boldsymbol{V}_{i,j} = 0$, the following update rules can be observed:

$$\boldsymbol{U}_{i,j} \leftarrow \boldsymbol{U}_{i,j}\frac{(\boldsymbol{X}\boldsymbol{V})_{i,j}}{(\boldsymbol{U}\boldsymbol{V}^\top\boldsymbol{V})_{i,j}}, \tag{10}$$

$$\boldsymbol{V}_{i,j} \leftarrow \boldsymbol{V}_{i,j}\frac{(\boldsymbol{X}^\top\boldsymbol{U})_{i,j}}{(\boldsymbol{V}\boldsymbol{U}^\top\boldsymbol{U} + \eta'\alpha(\boldsymbol{B} + \alpha\boldsymbol{V}))_{i,j}}. \tag{11}$$

2) Updates of α_s^f: With $\boldsymbol{U}_s^f$ and $\boldsymbol{V}_s^f$ calculated and fixed, we update α (i.e., α_s^f for each view f and class s) by solving the following problem:

$$\min_\alpha \sum_{f=1}^{F} \|\alpha_s^f \boldsymbol{V}_s^f\|_F^2, \quad \text{s.t} \quad \sum_{f=1}^{F} \alpha_s^f = 1, \alpha_s^f \geq 0, \tag{12}$$

It can be seen that (12) is a quadratic programming problem and thus can be solved using existing techniques. However, a trivial solution of such problems might be easily obtained:

$$\alpha_s^f = \begin{cases} 1, & \text{if } f = \arg\min_f \|\boldsymbol{V}_s^f\|_F^2 \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

The above solution suggests the coefficient matrix of class s to be dominated by the feature combination which produce the smallest $\boldsymbol{V}_s^f$ (i.e., non-sparse coefficients for combining the features for this class and thus lack discriminative abilities). In order to avoid this problem, we need to impose an addition regularization term and reformulate (12) as follows:

$$\begin{aligned} & \min_\alpha \sum_{f=1}^{F} \|\alpha_s^f \boldsymbol{V}_s^f\|_F^2 + \lambda H(\boldsymbol{V}_s^f)\alpha_s^f, \\ & \text{s.t} \quad \sum_{f=1}^{F} \alpha_s^f = 1, \alpha_s^f \geq 0. \end{aligned} \tag{14}$$

Based on (14), an image i is assigned to basis l if the lth entry of row i (in $\boldsymbol{V}_s^f$) is the maximum among all elements in that row. We note that, $H(\boldsymbol{V}_s^f)$ in (14) observes the entropy of the coefficients associated with view f:

$$H(\boldsymbol{V}_s^f) = -\sum_{l=1}^{L} \frac{n_l}{N} log_2 \frac{n_l}{N}, \tag{15}$$

where n_l is the number of images assigned to basis l. By minimizing $H(\boldsymbol{V}_s^f)$, our algorithm prefers the dominant features for particular bases, instead of equal weights which lack discriminative capabilities. Therefore, the separation between the coefficient matrices of different classes can be preserved.

Once the weights α_s^f for each view of class s is obtained, we update the coefficient matrix $\boldsymbol{V}_s^*$ for this class using (3), which ends the optimization for this class. Finally, we repeat this procedure for all classes for completing this iteration.

Algorithm 2 Multi-View NMF Algorithm

```
Input: Data matrices from $F$ views $\{\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots, \boldsymbol{X}^F\}$, number of bases $L$
Output: Basis matrices $\{\boldsymbol{U}_1^1, \boldsymbol{U}_1^2, \ldots, \boldsymbol{U}_S^F\}$, view weights $\{\alpha_1^1, \alpha_1^2, \ldots, \alpha_S^F\}$, coefficient matrices $\{\boldsymbol{V}_1^1, \boldsymbol{V}_1^2, \ldots, \boldsymbol{V}_S^F\}$ and $\{\boldsymbol{V}_1^*, \boldsymbol{V}_2^*, \ldots, \boldsymbol{V}_S^*\}$
Initialization by Alg.1
while (3) not converged do
    for $s = 1$ to $S$ do
        $\eta' \leftarrow \eta \sum_{j \neq s} \|\boldsymbol{V}_j^*\|_F^2$
        for $f = 1$ to $F$ do
            while (6) not converged do
                Updating $\boldsymbol{U}_s^f$ and $\boldsymbol{V}_s^f$ by (10), (11)
            end while
        end for
        Updating $\alpha_s^f$ for all views by solving (14)
    end for
end while
```

IV. Experimental Results

A. Caltech-101 Dataset

1) Dataset and Settings: We first consider the Caltech-101 dataset [21] for evaluating the clustering performance. In particular, we choose seven classes (i.e., Dollar-Bill, Faces, Garfield, Motorbikes, Snoopy, Stop-sign, and Windsor-chair) considered in [6], which also addressed clustering problems. Following the setting of [6], there are 31 to 100 images per category, and the total number of images are 431.

For each image, we extract the features of LBP, Gabor, and SIFT. When determining LBP for each pixel, we consider 8 uniformly sampled points on a circle (with the radius equal to 7 pixels) in a grid of 14 × 14 pixels, and we quantize the corresponding LBP values into a histogram with 256 bins as the final feature. We consider 7 scales in 8 different orientations for calculating the Gabor features. As for SIFT, we construct the bag-of-words (BoW) features for each image, while the codebook is constructed by K-means clustering with 256 codewords. We set $\eta = 10^{-5}$ in (3), and $\lambda = 15$ in (14).

2) Discussions: We compare our method with two baseline clustering algorithms: K-means clustering (KM) and standard NMF. For more advanced approaches, we specifically consider multi-view clustering algorithms of MultiNMF [19] and Robust Multi-View K-Means (RMKMC) [7]. We do not consider the approach of [6], since it is a graph-based approach (compared to other KM or NMF-based ones). We note that, when applying KM and NMF for clustering multi-view features, we first normalize the feature in each view before concatenating them into a single feature vector. For KM-based approaches of KM and RMKMC, the number of bases is set to the number of classes (i.e., 7). As for NMF-based methods (NMF, MultiNMF, and ours), we fix the number of bases $L = 22$. Later we will vary this L value to discuss the associated performance sensitivity. To perform the final clustering for the above NMF-based algorithms, a final K-means clustering with $K = 7$ clusters/classes will be applied on the observed coefficients for determining the final clustering results.

To perform quantitative evaluation, we consider the metrics of clustering accuracy (AC) and normalized mutual information (NMI) [15], [17]. Table I compares the performances of

TABLE I. CLUSTERING PERFORMANCE ON CALTECH-101.

Method	AC (%)	NMI (%)
K-means	52.41 ± 4.32	36.75 ± 4.19
NMF	55.22 ± 5.22	44.50 ± 4.42
MultiNMF [19]	63.67 ± 4.72	55.66 ± 4.48
RMKMC [7]	56.75 ± 3.93	45.77 ± 3.53
Ours	$\mathbf{75.19 \pm 8.61}$	$\mathbf{75.28 \pm 6.99}$

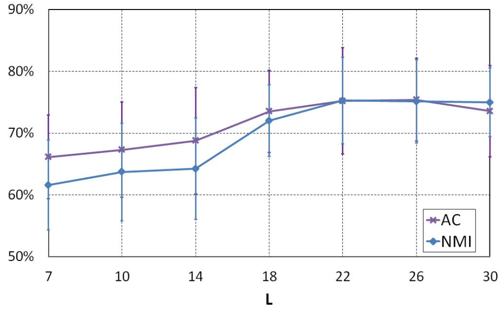


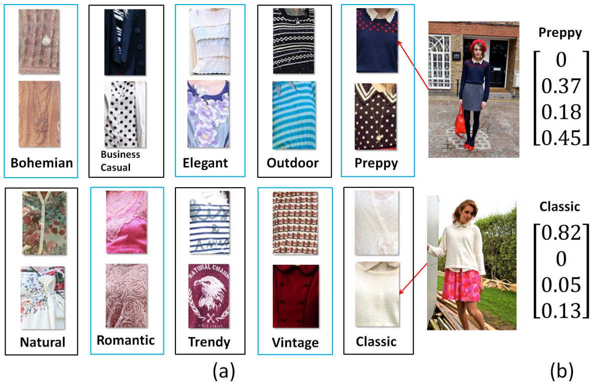
Fig. 3. Clustering performance of our method with different L values (i.e., number of bases in $\boldsymbol{U}$) on Caltech-101.

different clustering algorithms. It can be seen that, KM and NMF did not achieve satisfactory results since they integrate multiple features by simple feature concatenation. While MultiNMF, RMKMC, and our method advanced late fusion strategy to determine proper weights for each feature, we achieved the best performance among all due to the introduction of additional discriminating ability to our proposed NMF.

To discuss the performance sensitivity of NMF-based approaches to the value of L (i.e., the number of bases to be observed), we further conduct experiments by varying L and show the results in Figure 3. From this figure, we see that the performances in terms of both AC and NMI using different L were all above 60%, and thus the choice of L was not critical for our approach. However, we note that L should be at least larger than the number of categories to be clustered (i.e., 7), and it needs to satisfy $L < min(M_f, N_s)$ for the NMF requirements (recall that M_f is the feature dimension in view f, and N_s is the number of instances of class s). Therefore, we only show the results of L ranging from 7 to 30 in Figure 3.

B. Clothing Image Dataset

1) Dataset and Settings: To evaluate our proposed algorithm on clothing images, we consider image data with category information from Chictopia [1], which is an international online fashion site. We collect a total of 511 clothing images of ten different style categories (about 50 images per category). The ten style categories of interest are: Bohemian, Classic, Elegant, Natural, Trendy, Business Casual, Outdoor, Preppy, Romantic, and Vintage (see Figure 4(a) for examples). Since we focus on characterization of upper-body clothing images only, we manually crop out the upper-body parts of images as the regions of interest (ROI). For each ROI of an image, we extract four types of features: CIE-lab, LBP, Gabor, and SIFT (in terms of BoW models). When calculating CIE-lab features, we quantize an image into a three-dimensional histogram, where the numbers of bins for each channel are l = 2, a = 20, and b = 20, respectively (and thus a 800-dimensional vector will be formed). For other features, the settings are the same as those applied for Caltech-101. We have $\eta = 10^{-4}$ and $\lambda = 15$ for our proposed algorithm.



(a) (b)

Fig. 4. Clothing image dataset from Chictopia: (a) Example images of 10 image styles and (b) weights α_s^f learned by our multi-view NMF for the four features (in the order of CIE-lab, LBP, Gabor, and SIFT) of the two styles.

TABLE II. CLUSTERING PERFORMANCE ON THE CLOTHING IMAGE DATASET.

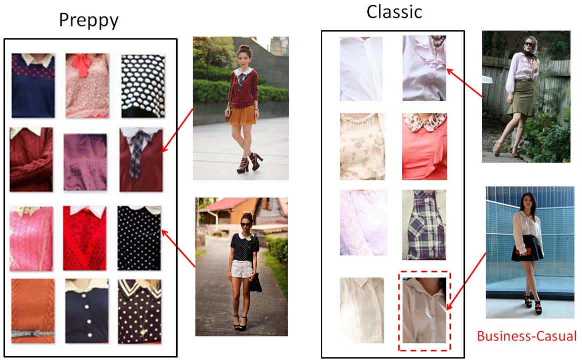
Method	AC (%)	NMI (%)
K-means	20.08 ± 0.62	11.53 ± 0.60
NMF	20.30 ± 1.10	10.92 ± 1.04
MultiNMF[19]	20.99 ± 1.71	11.13 ± 1.46
RMKMC[7]	20.12 ± 0.91	12.02 ± 0.74
Ours	$\mathbf{51.09 \pm 4.41}$	$\mathbf{44.63 \pm 3.41}$

2) Discussions: We randomly select 25 images per category for performing clustering. We repeat this process 20 times and report the average results. Table II compares the performances of different approaches, and it is clear that our proposed method outperformed baseline or state-of-the-art KM or NMF-based approaches. Since the styles of clothing images are typically more difficult to define, the AC/NMI results of all methods were generally lower than those reported for Caltech-101. We also note that, we fix the number of bases $L = 10$ in all NMF-based methods (including ours) when comparing the results. Similar to the remarks made for Caltech-101, our performance was generally not sensitive to the choice of this parameter. Figure 4(b) shows the feature weights learned for two selected clothing image styles.

Figure 5 shows the example clustering results of our proposed method. In Figure 5(a), we can see that several clothing images of Preppy were successfully identified as the same style category using our proposed algorithm. On the other hand, Figure 5(b) shows example results of visually similar clothing images grouped into the same cluster. However, since these images were assigned different style labels, this would be considered as an incorrect clustering result. This again supports the above comments on the challenge of determining proper style information for clothing images.

To verify the discriminative ability of our multi-view NMF, we provide additional clustering results using a single type of features in Figure 6, which compares AC performances of different methods using different types of features. In addition, by comparing Table II and Figure 6, it can be seen that the differences between standard KM/NMF-based approaches using single and concatenated features were not significant. This implies that successful clustering was not able to be

[1] http://www.chictopia.com/



(a) (b)

Fig. 5. Examples of (a) successful and (b) incorrect clustering results on the clothing image dataset.

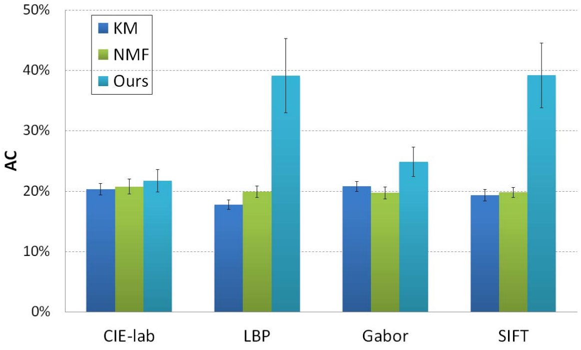


Fig. 6. AC comparisons of different methods on the clothing image dataset using a single type of features.

achieved by simple concatenation of multiple features. The same remarks can be made for NMI performance comparison (which we do not present here due to space limit).

C. Remarks on Image Classification

Finally, Table III lists the image classification results (via 4-fold cross-validation using linear SVMs) on the two image datasets. In addition to the use of single types of features, the baseline approach in Table III indicates simple concatenation of normalized features for classification. From Table III, we see that our use of $\boldsymbol{V}^*$ as image features were able to achieve the best recognition rates among all, and this further confirms the use of proposed method in characterizing images for solving both clustering and categorization problems.

V. Conclusion

In this paper, we proposed a multi-view NMF algorithm for characterizing images of different style categories, so that clustering and categorization tasks can be solved accordingly. Our multi-view NMF not only learns basis functions for describing images using different visual features, it also derives an optimal combination for integrating such features with discrimination guarantees. Therefore, our method is very different from prior early/late fusion strategies, which either perform simple feature concatenation, or focus on learning the weights for combining existing features without the ability for updating the feature representations. Empirical results on two image datasets verified the use of our proposed method on solving both image clustering and classification problems. We showed that our multi-view NMF outperformed baseline or state-of-the-art KM or NMF-based approaches, whether single or multiple types of features were considered.

TABLE III. Classification using different features.

Dataset	CIE-lab	LBP	Gabor	SIFT	Baseline	Ours
Caltech-101	–	31.55	37.12	38.52	33.18	**42.23**
Clothing	12.5	11.69	11.21	12.1	13.31	**25.40**

Acknowledgment

This work is supported in part by the Ministry of Science and Technology of Taiwan via NSC102-3111-Y-001-015 and NSC102-2221-E-001-005-MY2.

References

[1] Y. Kalantidis, L. Kennedy, and L.-J. Li, "Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos," *ICMR*, 2013.

[2] C. Snoek, M. Worring, and A. Smeulders, "Early versus late fusion in semantic video analysis," *ACM Multimedia*, 2005.

[3] G. Tzortzis and C. Likas, "Multiple view clustering using a weighted combination of exemplar-based mixture models," *IEEE TNN*, 2010.

[4] X. Chen et al., "Tw-k-means automated two-level variable weighting clustering algorithm for multi-view data," *IEEE TKDE*, 2013.

[5] B. Long, P. S. Yu, and Z. Zhang, "A general model for multiple view unsupervised learning," *SDM*, 2008.

[6] X. Cai et al., "Heterogeneous image feature integration via multi-modal spectral clustering," *IEEE CVPR*, 2011.

[7] X. Cai, F. Nie, and H. Huang, "Multi-view k-means clustering on big data," *IJCAI*, 2013.

[8] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, 1999.

[9] Y.-X. Wang and Y.-J. Zhang, "Nonnegative matrix factorization: a comprehensive review," *IEEE TKDE*, 2013.

[10] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, 2007.

[11] R. Sandler and M. Lindenbaum, "Nonnegative matrix factorization with earth movers distance metric for image analysis," *IEEE PAMI*, 2011.

[12] M. Gupta and J. Xiao, "Non-negative matrix factorization as a feature selection tool for maximum margin classifier," *IEEE CVPR*, 2011.

[13] Q. Gu et al., "Collaborative filtering: weighted nonnegative matrix factorization incorporating user and item graphs," *SDM*, 2010.

[14] W. Xum, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," *SIGIR*, 2003.

[15] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using l21-norm," *CIKM*, 2011.

[16] D. Cai, X. He, J. Han, and T. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE PAMI*, 2011.

[17] H. Liu, Z. Wu, X. Li, D. Cai, and T. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE PAMI*, 2012.

[18] D. Greene and P. Cunningham, "A matrix factorization approach for integrating multiple data views," *ECML PKDD*, 2009.

[19] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," *SDM*, 2013.

[20] I. Ramires, P. Sprechmann, and G. Spairo, "Classification and clustering via dictionary learning with structured incoherence and shared features," *IEEE CVPR*, 2010.

[21] L. Fei-Fei et al., "Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories," *IEEE CVPR WS Generative-Model Based Vision*, 2004.