Discriminative Basis Selection using Non-negative Matrix Factorization

Aruna Jammalamadaka, Swapna Joshi, S. Karthikeyan, B.S. Manjunath*
Dept. of Electrical and Computer Engineering, University of California, Santa Barbara CA 93106
{*arunaj, swapnaj, karthikeyan, manj*}*@ece.ucsb.edu*

Abstract

Non-negative matrix factorization (NMF) has proven to be useful in image classification applications such as face recognition. We propose a novel discriminative basis selection method for classification of image categories based on the popular term frequency-inverse document frequency (TF-IDF) weight used in information retrieval. We extend the algorithm to incorporate color, and overcome the drawbacks of using unaligned images. Our method is able to choose visually significant bases which best discriminate between categories and thus prune the classification space to increase correct classifications. We apply our technique to ETH-80, a standard image classification benchmark dataset. Our results show that our algorithm outperforms other state-of-the-art techniques.

1. Introduction

There are two main approaches to the challenging problem of image classification: object-based and scene-based. Matrix factorization is an example of a prevalently used scene-based classification method. There are many well established matrix factorization algorithms, such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Non-negative matrix factorization (NMF) [2]. These methods all learn to represent data as a linear combination of basis images; however, each algorithm factorizes the input into these basis vectors subject to different constraints. We introduce a method for Discriminative Basis Selection (DBS) using image bases arrived at by Localized NMF (LNMF) [4]. Our subset selection algorithm is based on the popular term frequency-inverse document frequency (TF-IDF) [5] weighting scheme.

PCA, a popular factorization technique, computes orthogonal basis images that lie in the direction of largest variance. The linear combination of the basis components can be either additive or subtractive and involve complex cancellations between these vectors. A drawback of this is that the basis representation of the features lack intuitive visual understanding. NMF overcomes this drawback by using non-negativity constraints. This leads to visually significant bases, or non-negative parts of an image. Studies have shown that there is physiological and psychological evidence for parts based representation, such as the one NMF produces, in the human brain [2] and hence NMF has received considerable attention in recent years.

There have been many variants of NMF proposed in recent years. For example, LNMF guarantees a localized parts-based representation by adding further constraints to enforce spatial locality of bases. However, it does not encode discrimination information for a classification problem and this can lead to redundant bases. Another variant which is specifically meant for classification is Fisher NMF (FNMF) [7]. This method incorporates Fisher constraints within the NMF framework in order to boost classification results. In Weighted NMF (WNMF) [1] weights are assigned to images inverse to the probability of that image appearing in the training set. The aim is to get fewer redundant bases in order to improve the distinction between classes. The drawback of this method is that one must assume the probability of each training image ocurring in the training set will also apply to the test images.

Influenced by both WNMF and [6], where video keyframes are ranked using TF-IDF in order to retrieve similar objects from feature-length films, we propose a new method called DBS. Our contributions are as follows: 1) A novel method for image classification using LNMF on unaligned and color images 2) A basis subset selector using a scheme similar to TF-IDF which discards bases that occur frequently across image categories.

The rest of the paper is organized as follows: Section 2 describes LNMF, the TF-IDF weighting scheme and the ETH-80 dataset. Section 3 describes our methodol-

*Thanks to NSF Grants ITR-0331697, and NSF III-0808772.

ogy in detail. Section 4 gives a detailed explanation of our results, and finally Section 5 concludes our paper and describes possible further applications and work.

2. Review

2.1. LNMF

Similar to PCA, and regular NMF, LNMF decomposes a non-negative matrix ($\mathbf{V}$) to a set of non-negative basis ($\mathbf{W}$) and corresponding non-negative coefficients ($\mathbf{H}$),

$$\mathbf{V}_{n\times n_t} \approx \mathbf{W}_{n\times m}\mathbf{H}_{m\times n_t} \tag{1}$$

where $\mathbf{V} = [v_{ij}] = [\mathbf{v}_1, ..., \mathbf{v}_{nt}]$ is a n x n_t matrix , n is the total number of pixels in each image, $\mathbf{v}_j$ is the jth input image represented as a column vector, and n_t is the number of training images. We denote the basis matrix $\mathbf{W} = [w_{ij}] = [\mathbf{w}_1, ..., \mathbf{w}_m]$ as an n x m matrix (where $m < n$ is the number of bases). Every column of $\mathbf{V}$ is a weighted sum of every row of $\mathbf{W}$ where the corresponding column in $\mathbf{H} = [h_{ij}] = [\mathbf{h}_1, ..., \mathbf{h}_{nt}]$ are the weights.

Like NMF, this factorization is achieved by minimizing the divergence between $\mathbf{V}$ and $\mathbf{WH}$ with the constraints that both should be non-negative. However LNMF has 3 additional constraints. The first is that it attempts to minimize the number of basis components required to represent V. The second is that it tries to make the bases as orthogonal as possible and the third is that only bases containing the most important information should be retained. This is done by maximizing the total sum of squared projection coefficients.

The constrained divergence between $\mathbf{V}$ and $\mathbf{WH}$ is defined as [4]

$$D(\mathbf{V}||\mathbf{WH}) = \sum_{i,j}\left(v_{ij}\log\frac{v_{ij}}{y_{ij}} - v_{ij} + y_{ij}\right) + \alpha\sum_{i,j}u_{ij} - \beta\sum_i q_{ii}, \tag{2}$$

where $\mathbf{WH} = \mathbf{Y} = [y_{ij}], \alpha, \beta > 0$ are constants, $\mathbf{W^TW} = \mathbf{U} = [u_{ij}]$ and $\mathbf{HH^T} = \mathbf{Q} = [q_{ij}]$

LNMF tries to mimic the way humans perceive visual information as a composite of simpler objects. Its basis vectors contain localized features that correspond better with intuitive notions of the parts of the images.

2.2. TF-IDF

TF-IDF [5] is a term weighting method popularly used in document clustering, information retrieval and text mining. It is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance of each word-document pair increases proportionally to the number of times that word appears in that document but is offset by the frequency of the word in the corpus. The term frequency matrix is calculated as shown:

$$\mathrm{tf_{pq}} = \frac{n_{pq}}{\sum_k n_{rq}} \tag{3}$$

where n_{pq} is the number of occurrences of the considered term t_p in document d_q, and the denominator is the sum of number of occurrences of all terms in document d_q. The inverse document frequency vector is computed as shown:

$$\mathrm{idf_p} = \log\frac{|D|}{|\{d : t_p \in d\}|} \tag{4}$$

with $|D|$: total number of documents in the corpus $|\{d : t_p \in d\}|$: number of documents where the term t_p appears (that is $n_{pq} \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to use $1 + |\{d : t_p \in d\}|$ The TF-IDF value for a term will always be greater than or equal to zero.

Then $(\text{tf-idf})_{\mathrm{pq}} = \mathrm{tf_{pq}} \times \mathrm{idf_p}$. A high weight in TF-IDF is reached by a high term frequency in the given document and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms and are then used to rank documents for search and retrieval.

2.3. Dataset

We used the ETH-80 Dataset from ETH Zurich [3]. It contains 80 objects from 8 categories (apple, car, cow, cup, dog, horse, pear and tomato). There are 10 objects per category that span large in-class variations while still clearly belonging to the category. Each object is represented by 41 images from viewpoints spaced equally over the upper viewing hemisphere at distances of 22.5 to 26 degrees (See Figure 1). This allows to analyze the performance of different recognition methods not only from a 1D circle or a few canonical viewpoints, but from multiple viewing positions. We use 5 views of each object (every 8th view) from the first 7 objects for training and 5 views from the remaining 3 objects in each class for testing. This gives us 35 training images and 15 testing images for each object class.

3. Our Approach

The theory at the heart of DBS is that image bases found using Non-negative Matrix Factorization (NMF)

Figure 1: ETH-80 Dataset (Image best viewed in color.)

can be thought of as parts which, when summed, create the whole image. This is comparable to the way in which a combination of words create an entire document, and it allows us to make use of the TF-IDF concept.

First we normalize and vectorize all the images in the dataset. Then we train each category separately with LNMF. This is in order to obtain category specific bases which we can then prune to contain only the bases which best describe that category's object. We then concatenate the bases from each category and reproject all the training images onto the full set of bases. We call this Category LNMF (CLNMF) in order to compare to our classification method later.

To use TF-IDF weights in the image context we treat LNMF bases as words and image categories as documents. The coefficients in $\mathbf{H}$ give us n_{pq}. This is a natural choice because each h_{ij} describes the amount of basis i which is needed to reconstruct image j. To create the TF matrix we normalize each column of $\mathbf{H}$ by the sum of the coefficients of each image so that they sum to one. This is analogous to taking the number of occurences of a word and dividing by the total number of words in a document. At this point we use the mean TF value across each category because we are interested in how the bases distinguish between categories, rather than within them. Now that we have a term frequency for each basis in each category we can compute the IDF.

To create the IDF we must define a threshold which signifies the "existence" of a basis in a given image. For this we use $1/m$ because our coefficients after normalization can now be thought of as a probability distribution. If the reconstruction coefficient is higher than this number it implies that the basis is more useful than in the uniform case. Using this threshold we can now compute the full TF-IDF as shown below:

$$\mathrm{tf}_{\mathrm{image_{ij}}} = \frac{h_{ij}}{\sum_i h_{ij}} \tag{5}$$

where h_{ij} is the reconstruction coefficient corresponding to basis $\mathbf{w}_i$ in image $\mathbf{v}_j$, and the denominator is the sum of all reconstruction coefficients in image $\mathbf{v}_j$.

$$\mathrm{tf}_{\mathrm{category_{ik}}} = \frac{\sum_{j=1}^{n_k} tf_{image_{ij}}}{n_k} \tag{6}$$

where n_k is the number of images in the kth category.

$$\mathrm{idf_i} = \log \frac{|K|}{|\{k : \mathbf{w}_i \in k\}|} \tag{7}$$

with $|K|$: total number of categories in the training set (in our case $K = 8$). $|\{k : \mathbf{w}_i \in k\}|$: number of categories where the basis $\mathbf{w}_i$ reconstructs more than $1/m$ of that category k.

Now in order to compute a weight for each basis we again take a mean across all categories. This vector of basis weights represents the usefulness of each basis in classification. At this point we threshold once again to reduce the dimensionality of the classification space. This threshold is empirically found.

Furthermore, incorporating color helps discriminate between objects with similar structures like apples and pears. NMF involves a matrix multiplication therefore we can consider each color component of the vectorized image to be factorized separately. Thus after the factorization is done we split the color channels apart and perform three TF-IDF computations; then we concatenate the color channels for classification. For an example of a single-channel basis see Figure 2. As seen in Section 4 blue bases corresponding to the background of the training images is eliminated.

4. Experimental Results

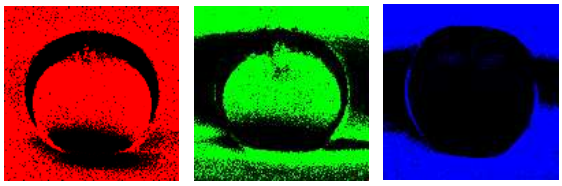
Our main goal is to perform image classification with the ETH-80 benchmark dataset (see Figure 1). We present the performance of our proposed algorithm on the dataset and compare our algorithm against three other methods: FNMF [7], LNMF [4], and CLNMF as explained in Section 3. For classification we use k-neareast neighbors where $k = 3$.

FNMF is shown to work better for classification purposes than LNMF [7]; however, as we can see from Table 1, our method outperforms FNMF for all but one of the chosen numbers of bases. It should be noted that this is a skewed comparison because while FNMF is using all 96 bases our method is using only 82 to obtain the same accuracy. We have also tested FNMF using the number of chosen bases and have found the results to be much worse than those of our method. We believe that there are two main reasons for this result. Firstly, FNMF does not prune the classification space in a way that specifically de-emphasizes those bases which do not differentiate between categories well. Secondly,

FNMF, and NMF in general, do not do well on unregistered images. This is because when learning visual bases an unregistered training set leads to an overfitting of the training data. We show here that our subset selection method overcomes this problem as well.

LNMF is also outperformed by our method since it does not encode any type of discrimination information for a classification problem and this can lead to redundant bases. Our method aims to eliminate these redundant bases and compact the classification space such that only the most discriminative bases are retained. As we notice in Figure 2 the bases with the highest weights are not from the blue channel. This is expected because the blue background is common to all images and has therefore been discarded while the bases capturing the objects are preserved, giving a good illustration of the advantage of retaining color as a feature.

In [8] the authors attempt to eliminate bases corresponding to noise and retain the bases which truly discriminate the data. The bases are arrived at through LDA and then pruned by finding each eigenvector's correlation to the range of the input image matrix $\mathbf{V}$. A lack of correlation implies the eigenvector can be eliminated from consecutive computations as it does not carry any discriminant information. Although this paper uses more objects for training than we do (9 versus our 7) they report a maximum classification rate of 76.05% while we achieve 80.0%.



(a) Red Channel (b) Green Channel (c) Blue Channel

Figure 2: The highest to lowest weighted bases for apple with corresponding weights (a.) 0.0559, (b.) 0.0578 and (c.) 0.0005. (Image best viewed in color.)

5. Conclusion

We have presented a novel method to use TF-IDF weights for image bases computed by LNMF in a way which boosts classification rates compared to current methods. We have found that much in the same way TF-IDF weights are low for words which occur commonly in the corpus, such as "the" or "of", bases with low weights in our scheme correspond to parts of the image which are shared among categories, such as the background. Besides this our further contributions include a structured methodology for using NMF with color and unaligned images and thus generally improving the usage of NMF for object classification. In the future we aim to discover further comparisons to concepts in the document clustering domain such as synonymy and polysemy. We are currently exploring extensions to our algorithm to incorporate spatial correlations within the image which may lead to segmentation of meaningful parts.

Table 1: Average recognition rate (in percentage) with varying the number of bases. Bold numbers represent the highest recognition rate for each method.

Methods	Number of bases					
	48	64	80	96	112	240
FNMF	69.2	68.3	72.5	73.3	**75**	74.3
LNMF	69.2	**75**	72.5	72.5	70	70
CLNMF	64.2	63.3	64.2	64.2	65.8	**66.7**
	Number of bases chosen					
	32	45	56	82	79	204
DBS	74.2	77.5	75	73.3	**80.0**	78.3

References

[1] D. Guillamet, J. Vitria, and B. Schiele. Introducing a weighted non-negative matrix factorization for image classification. *Pattern Recognition Letters*, 24(14):2447–2454, 2003.

[2] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[3] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2. Citeseer, 2003.

[4] S. Li, X. Hou, H. Zhang, and Q. Cheng. Learning spatially localized, parts-based representation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1. IEEE Computer Society; 1999, 2001.

[5] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval* 1. *Information processing & management*, 24(5):513–523, 1988.

[6] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, volume 2, pages 1470–1477. Citeseer, 2003.

[7] Y. Wang, Y. Jia, C. Hu, and M. Turk. Non-negative matrix factorization framework for face recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 19(4):495–512, 2005.

[8] M. Zhu. Pruning noisy bases in discriminant analysis. *IEEE Transactions on Neural Networks*, 19(1):148–157, 2008.