

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/2530070>

Analyzing Non-Negative Matrix Factorization for Image Classification

Article · May 2002

Source: CiteSeer

CITATIONS

36

READS

272

3 authors, including:



Jordi Vitrià

University of Barcelona

236 PUBLICATIONS 3,009 CITATIONS

SEE PROFILE

Analyzing Non-Negative Matrix Factorization for Image Classification*

David Guillamet¹, Bernt Schiele² and Jordi Vitrià¹

¹Computer Vision Center, Dept. Informàtica
Universitat Autònoma de Barcelona
08193 Bellaterra, Barcelona, Spain
{davidg,jordi}@cvc.uab.es

²Perceptual Computing and Computer Vision
Group, ETH Zurich
CH-8092 Zurich, Switzerland
{schiele}@inf.ethz.ch

Abstract

The Non-negative Matrix Factorization technique (NMF) has been recently proposed for dimensionality reduction. NMF is capable to produce a region- or part-based representation of objects and images. This paper experimentally compares NMF to Principal Component Analysis (PCA) in the context of image patch classification. A first finding is that the two techniques are complementary and that their respective performance is correlated to the within class scatter. This paper also analyses different techniques to combine these complementary methods. In the first combination scheme the best technique for each class is chosen and the results are merged. The second combination scheme builds a hierarchy of classifiers where again for each classification task the best technique is chosen. Additionally, incorporation of the classification results of neighboring image patches further improves the overall results.

1. Introduction

Principal Component Analysis (PCA) is a very popular technique for dimensionality reduction. It is a well-known fact that PCA is optimal only with respect to the reconstruction error and not for the separation and recognition of classes. Nevertheless, PCA is often used directly for pattern and object recognition tasks. In the computer vision community for example it has been used for recognition of faces [7] and 3D objects [6], or dealing with partial occlusions by using robust estimation techniques [1, 2].

Recently, Lee and Seung [3] proposed a new technique, called Non-negative Matrix Factorization (NMF), to obtain

a reduced representation of data. NMF differs from other methods by its use of non-negativity constraints. They demonstrated with a set of face images [3] that NMF can be used to obtain a basis of localized features in an unsupervised way. Many of those localized features correspond to the intuitive notion of face parts such as eyes and mouth.

This paper presents a comparative study of both techniques (PCA and NMF) in a color classification scheme. The main goal is to test and present NMF as a relevant classifier that can be used instead or in conjunction with PCA. The main motivation of introducing NMF in such a framework is because our data is positively defined and NMF is based on positive restrictions, meaning that NMF can be a suitable technique for such a problem. We analyze both techniques in a 10 class problem and we show that PCA and NMF have complementary advantages for classification. Furthermore, both techniques are combined into a common classifier improving the results of each technique alone. The results can be further improved combining both techniques in a hierarchical fashion choosing the best technique at each level of the hierarchy. We also investigate when and why PCA or NMF are more appropriate for classification depending on the class distribution.

2. PCA and NMF techniques

Principal Component Analysis (PCA): Due to the high dimensionality of data, similarity and distance metrics are computationally expensive and some compaction of the original data is needed. Principal Component Analysis is an optimal linear dimensionality reduction scheme with respect to the mean squared error (MSE) of the reconstruction. For a set of N training vectors $X = \{x^1, \dots, x^N\}$ the mean ($\mu = \frac{1}{N} \sum_{i=1}^N x^i$) and covariance matrix ($\Sigma = \frac{1}{N} \sum_{i=1}^N (x^i - \mu)(x^i - \mu)^T$) can be calculated. Defining a projection matrix E composed of the K eigenvectors of Σ with highest eigenvalues, the K -dimensional representation of an original, n -dimensional vector x , is given by the projection $y = E^T(x - \mu)$.

*This work was supported by IST project IST-1999-20188-CORKINSPECT, sponsored by the European Commission and by Comissionat per a Universitats i Recerca de la Generalitat de Catalunya and Ministerio de Ciencia y Tecnología grant TIC2000-0399-C02-01. The Corel database used in this paper has been financed by the CogVis project, funded in part by the Commission of the European Union under contract IST-2000-29375, and the Swiss Federal Office for Education and Science (BBW 00.0617).

Non-Negative Matrix Factorization (NMF): NMF is a method to obtain a representation of data using non-negativity constraints. These constraints lead to a part-based representation because they allow only additive, not subtractive, combinations of the original data [3]. Given an initial database expressed by a $n \times m$ matrix V , where each column is an n -dimensional non-negative vector of the original database (m vectors), it is possible to find two new matrices (W and H) in order to approximate the original matrix $V_{i\mu} \approx (WH)_{i\mu} = \sum_{a=1}^r W_{ia}H_{a\mu}$. The dimensions of the factorized matrices W and H are $n \times r$ and $r \times m$, respectively. Usually, r is chosen so that $(n + m)r < nm$. Each column of matrix W contains a basis vector while each column of H contains the weights needed to approximate the corresponding column in V using the basis from W . In the PCA context, each column of matrix W represents an eigenvector and the factorized matrix of H represent the eigenprojections. In contrast to PCA, NMF does not allow negative entries in the factorized matrices W and H permitting the combination of multiple basis images to represent an object.

In order to estimate the factorization matrices, an objective function has to be defined. A possible objective function is given by $F = \sum_{i=1}^n \sum_{\mu=1}^m [V_{i\mu} \log(WH)_{i\mu} - (WH)_{i\mu}]$. This objective function can be related to the likelihood of generating the images in V from the basis W and encodings H . An iterative approach to reach a local maximum of this objective function is given by the following rules [3]: $W_{ia} \leftarrow W_{ia} \sum_{\mu} \frac{V_{i\mu}}{(WH)_{i\mu}} H_{a\mu}$, $W_{ia} \leftarrow \frac{W_{ia}}{\sum_j W_{ja}}$, $H_{a\mu} \leftarrow H_{a\mu} \sum_i W_{ia} \frac{V_{i\mu}}{(WH)_{i\mu}}$. Initialization is performed using positive random initial conditions for matrices W and H . The convergence of the process is also ensured. See [3, 4] for more information.

3. Experimental Results

In order to compare PCA and NMF experimentally, we have selected 932 color images from the Corel Image database. These images were selected in order to obtain data for 10 different classes of image patches namely: clouds, grass, ice, leaves, rocky mountains, sand, sky, snow mountains, trees and water. Image patches from each of those class contain different color tonalities rather than one unique and global color. Each image is automatically divided in 10×10 local regions or image patches of $3456(48 \times 72)$ RGB pixels. Whenever possible, each of these image patches is labeled according to the 10 classes mentioned above. Each image patch is represented using a color histogram of 512 dimensions (8 bins per color). 1000 color histograms are randomly selected for training and other 1000 for testing.

Experimental comparison of PCA and NMF This experiment compares both PCA and NMF techniques in terms

of recognition rates. Given a local color histogram to classify, we project it to all 10 models learned with one method (PCA or NMF) and choose the model that better reconstructs the original histogram. This reconstruction scheme is often used [5, 8]. To compare PCA and NMF, table (1) shows the confusion matrices obtained for classification. Overall, PCA is able to classify 56.43% and NMF 59.89% noticing an improvement.

Confusion Matrix for PCA									
	Model Clouds	Model Grass	Model Ice	Model Leaves	Model Rocky	Model Sand	Model Sky	Model Snow	Model Tree
Clouds	244	1	358	0	4	32	325	18	0
Grass	0	927	0	14	0	33	0	0	26
Ice	17	0	854	0	0	3	90	10	2
Leaves	0	250	31	662	0	10	0	0	31
Rocky	26	47	29	0	285	363	35	29	155
Sand	21	26	22	0	25	874	7	0	23
Sky	18	0	351	0	0	16	563	25	0
Snow	89	0	411	0	3	33	160	218	5
Tree	1	207	3	20	14	39	1	2	687
Water	19	10	439	1	3	37	131	13	18
Total Recognition Rate: 56.43 %									

Confusion Matrix for NMF									
	Model Clouds	Model Grass	Model Ice	Model Leaves	Model Rocky	Model Sand	Model Sky	Model Snow	Model Tree
Clouds	433	2	141	0	7	33	21	7	110
Grass	0	788	0	53	15	28	0	0	116
Ice	115	3	552	1	5	1	89	119	3
Leaves	5	87	18	793	5	15	2	8	62
Rocky	22	24	17	0	405	285	33	71	131
Sand	42	10	8	0	116	786	2	12	22
Sky	213	6	160	1	8	13	371	54	0
Snow	112	0	196	0	17	15	82	458	3
Tree	0	68	2	30	58	21	1	1	808
Water	28	3	163	7	34	12	46	79	33
Total Recognition Rate: 59.89 %									

Table 1. Confusion Matrix for PCA in 60D (above) and NMF in 60D (below) using the L2-norm of the retroprojected vectors.

Analyzing both confusion matrices of table (1) we can see that the employed data is difficult to separate. This is due to the large intersections and similarities of several classes. Interestingly, some classes are better classified using PCA whereas some are better classified NMF. This indicates that constructing a combined classifier of both techniques may lead to better classification results.

Dispersion of classes Before combining the two techniques we would like to analyze why and when PCA and NMF are more appropriate than the other technique. To do this we analyzed the data distribution of the different classes in the original space of color histograms. As an estimate of the elongation or compactness of the distribution of each class we can directly use the eigenvalues of the covariance matrix (Σ). More specifically we use $sum(trace(\Sigma))$. This is shown in table (2) where we can see that the sky class is the most dispersed class and the tree class is the most compact.

Class	Tree	Rocky	Grass	Snow	Leaves	Water	Clouds	Ice	Sand	Sky
Analysis	0.146	0.195	0.209	0.219	0.262	0.306	0.350	0.356	0.382	0.698

Table 2. Compactness estimation.

In order to compare PCA and NMF models for each in-

dividual class we took the confusion matrices in table (1) as a reference to obtain a measure of goodness (α_{CLASS}) of each model as defined in expression (1). The measure of goodness for each class is shown in table (3)

$$\alpha_{\text{CLASS}} = \frac{\text{Correct classification vectors of CLASS}}{\# \text{ of vectors of other classes classified as CLASS}} \quad (1)$$

Algorithm	α_{Clouds}	α_{Grass}	α_{Ice}	α_{Leaves}	α_{Rocky}	α_{Sand}	α_{Sky}	α_{Snow}	α_{Tree}	α_{Water}
PCA	1.2775	1.7135	0.5195	18.91	5.819	1.5442	0.7517	2.2474	2.6423	1.4622
NMF	0.8063	3.8818	0.7830	8.6196	1.52	1.8582	0.7860	1.0088	2.1838	1.2143

Table 3. α_{class} defined in expression (1) per each technique.

In table (3) the best technique (PCA or NMF) for each class is presented using bold face. For example, clouds are better classified using PCA than using NMF. Analyzing tables (2) and (3) we observe a high correlation between dispersed classes and the use of NMF. That is, PCA can be used to classify the more compact classes and NMF for the more dispersed and complex ones. In tables (2) and (3) we see that sky, sand, ice and grass classes are better classified with NMF and that three of these classes are also the most dispersed ones (sky, sand and ice). The only exception of this is class grass, which is probably due to the fact this class is relatively distinct with respect to the other nine classes.

Combination of models The above results suggest that PCA as well as NMF have their respective strengths which motivate the combination of both techniques in one classification scheme. As we will see in the following, the reconstruction error of both techniques are comparable which allows a direct combination of both techniques. PCA is optimal in terms of the reconstructed error whereas NMF is capable to represent data in terms of a non-negative basis. This means that the reconstructed error obtained by the NMF will not be optimal. An experiment was carried out in order to know whether the errors of both techniques are comparable. This experiment consists of learning a PCA and a NMF model for each class, to project all 1000 local color histograms of each class and to compare the reconstruction error obtained for each vector. Table (4) summarizes this experiment.

	Grass	Tree	Sand	Ice	Rocky	Leaves	Water	Snow	Clouds	Sky
Mean Error PCA	0.40	1.45	2.14	0.46	8.32	10.76	6.34	3.74	4.85	9.34
Mean Error NMF	5.16	3.63	13.67	5.88	21.65	21.61	22.47	14.47	18.34	43.59
PCA > NMF	904	765	717	692	683	578	563	541	522	439

Table 4. PCA and NMF reconstruction errors.

As expected, table (4) shows that the mean of the reconstruction error of PCA is less than the one obtained for NMF (first and second rows). The last row shows however, how many local regions have a reprojection error that

is smaller for PCA than for NMF. The sky class for example contains only 439 vectors that are better represented with PCA. This implies that some classes are better represented with PCA and some other with NMF. Overall we can say that the reconstruction errors of both techniques are comparable, which allows to merge both methods in a combined classifier.

The central idea for the first combination scheme is to choose the best method (PCA or NMF) for each class and directly use the reconstruction error to classify the test data. Table (5) shows the confusion matrix obtained with this combination scheme. Most interestingly the recognition rate 62.20% of this combined classifier is higher than the recognition rates of PCA or NMF alone. In this particular case, grass, ice, sand and sky models are represented with the NMF technique and the other classes are represented using PCA.

	Model Clouds	Model Grass	Model Ice	Model Leaves	Model Rocky	Model Sand	Model Sky	Model Snow	Model Tree	Model Water
Clouds	464	1	182	0	10	32	201	60	1	49
Grass	0	783	0	61	22	19	0	0	113	2
Ice	59	0	674	1	1	0	109	45	4	107
Leaves	2	81	7	825	7	0	6	0	1	44
Rocky	44	18	9	0	433	204	30	36	177	49
Sand	69	8	10	0	116	744	1	2	42	8
Sky	62	3	261	0	1	17	543	50	1	62
Snow	213	0	256	0	13	6	47	324	6	135
Tree	2	56	1	36	24	8	0	2	838	33
Water	55	4	185	2	17	12	39	67	27	592

Total Recognition Rate: 62.20 %

Table 5. Confusion Matrix for the combined classifier in 60D using the L2 of the retroprojected vectors.

The first combination scheme chooses the best technique for each class. We can also take into account that the data can be classified in a hierarchical fashion. For example we can separate blue classes against other classes and then refine this classification until the separation of two specific classes. This hierarchical scheme relies on the possibility that data can be separated in different steps and we can consider different kinds of classifiers for each level. The best distribution of classes and techniques for such a classification scheme is described in figure (1). Given a node in this new tree representation and having a local vector to classify between the right and left leaves, we obtain the retroprojected distance of this vector and we choose the leaf that contains the minimal distance. Under the name of each class in figure (1), the employed technique for each discrimination task is shown. Considering the hierarchical distribution of models shown in figure (1), we obtain the results presented in table (6) noticing again an improvement of the recognition rate of 64.03%.

Neighborhood analysis Until now, each testing vector has been classified separately. But we can take into account for example that a sky region is typically surrounded by

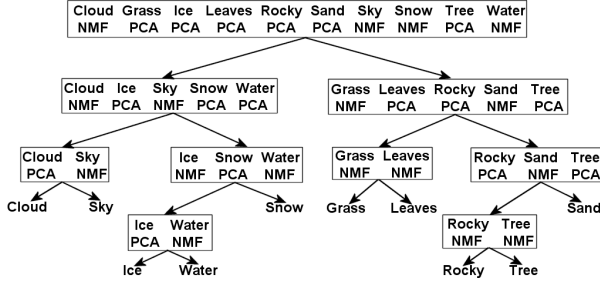


Figure 1. Optimal hierarchical representation to classify the initial 10 classes of data.

	Model Clouds	Model Grass	Model Ice	Model Leaves	Model Rocky	Model Sand	Model Sky	Model Snow	Model Tree	Model Water
Data Clouds	370	1	160	0	16	32	202	167	1	51
Data Grass	0	795	0	51	26	18	0	0	110	0
Data Ice	91	2	625	0	2	2	90	77	3	108
Data Leaves	0	91	2.2	809	6	6	0	8	49	9
Data Rocky	12	20	14	0	507	209	24	64	140	10
Data Sand	13	9	12	0	148	780	3	5	27	3
Data Sky	50	2	131	0	8	9	618	94	1	87
Data Snow	93	0	243	0	31	14	41	517	7	54
Data Tree	0	61	3	33	66	8	0	2	822	5
Data Water	19	4	173	7	35	15	26	122	39	560

Total Recognition Rate: 64.03 %

Table 6. Confusion Matrix for the combined hierarchical classifier in 60D using the L2 of the retroprojected vectors.

other local sky regions. That is, we can consider the neighborhood of each local region to improve the initial results and decrease the effect of outliers. Two different neighborhood analysis have been considered: Maximal neighborhood and mean neighborhood. Once we have all the local regions of a certain zone labeled according to one of the previous classifiers, we can analyze the neighborhood of each local region by searching for the most probable model for it. That is, if we have a region labeled as water but the entire neighborhood is labeled as sky, this local region will be finally labeled as sky. We have named this analysis as *maximal neighborhood*. Another technique, called *mean neighborhood*, is also considered for local neighborhood analysis. Since we have a region and its neighborhood with associated reconstructed distances we can consider all the neighborhood distances to obtain a mean distance for each local region. So, if we have a local region x , the mean distance that we will associate to it will be $\sum_{i=1}^N \text{dist}(x_i)/N$, where N is the number of local regions in the neighborhood.

By considering the neighborhood influence, we have to use all local regions instead of a random test set. Therefore in the following we consider the entire database of 45973 local regions and we repeated all the previous classification experiments in conjunction with this neighborhood techniques (using 8 connectivity). Results are shown in table (7). We should note that when using a hierarchical representation, the *mean neighborhood* analysis causes a bad influence because the results obtained are the same that when we use PCA lonely (noted as NO SENSE in the table). Best results are obtained using a PCA+NMF mixed classifier in

conjunction with both neighborhood techniques. The initial recognition rate 60.26% using PCA alone can be improved to a 75.94% using the combination of both PCA and NMF techniques.

Model Representation	Direct Implementation	Neighborhood Mean	Neighborhood Maximal	Neighborhood Maximal + Mean
PCA Representation	60.256	65.98	61.53	66.76
NMF Representation	63.80	67.98	67.76	69.64
PCA + NMF Mixed Representation	67.35	73.41	71.28	75.94
PCA + NMF Tree Representation	68.79	NO SENSE	69.74	NO SENSE

Table 7. Results using the entire database of 45973 local vectors.

4. Conclusions

In this paper we have experimentally analyzed an alternative technique to Principal Component Analysis (PCA), the so called Non-negative Matrix Factorization (NMF). NMF was initially tested with faces to obtain parts of faces but not used in a classification framework. The results of this paper demonstrate the applicability of NMF as a classification technique and compares the results to PCA. On the presented data NMF obtains better classification results than PCA alone. Since the two techniques are complementary they can be merged in a combined classifier thereby improving the initial classification results. The best classification results are obtained when we divide the original classification problem in an hierarchical fashion and select the best technique for each class on each level of the hierarchical analysis. We also apply a neighborhood analysis that again increases the recognition rates.

An interesting and important question is when and why PCA or NMF are more suited for the classification of certain classes. In this paper, we have experimentally pointed out that PCA can represent better those classes that are more compact and NMF is able to classify better the most dispersed and complex ones. It remains however a more comprehensive analysis in order to confirm these first findings.

References

- [1] M. Black and A. Jepson. Eigenttracking : Robust matching and tracking of articulated objects using a view-based representation. *IJCV*, 26(1):63–84, 1998.
- [2] F. de la Torre and M. Black. Robust principal component analysis for computer vision. In *Proc. of ICCV'2001*, volume 1, pages 362–369, 2001.
- [3] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.
- [4] D. Lee and H. Seung. Algorithms for non-negative matrix factorization. *NIPS*, 2000.
- [5] A. Leonardis and H. Bischof. Multiple eigenspaces by mdl. In *Proc. ICPR*, volume 1, pages 233–237, 2000.
- [6] H. Murase and S. Nayar. Visual learning and recognition of 3d objects from appearance. *IJCV*, 14:5–24, 1995.
- [7] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Neuroscience*, 3(1):71–86, 1991.
- [8] J. Winkeler, B. Manjunath, and S. Chandrasekaran. Subset selection for active object recognition. In *Proc of CVPR*, pages 511–516, 1999.