VECTOR QUANTIZATION WITH CONSTRAINED LIKELIHOOD FOR FACE RECOGNITION

Dimche Kostadinov, Sviatoslav Voloshynovskiy, Maurits Diephuis and Sohrab Ferdowsi

University of Geneva
Computer Science Department, Stochastic Information Processing Group
7 Route de Drize, Geneva, Switzerland

ABSTRACT

In this paper, we investigate the problem of visual information encoding and decoding for face recognition. We propose a decomposition representation with vector quantization and constrained likelihood projection. The optimal solution is considered from the point of view of the best achievable classification accuracy by minimizing the probability of error under a given class of distortions. The performance of the proposed model of information encoding/decoding is compared with the performance of those based on sparse representation. The computer simulation results confirm the superiority of the proposed vector quantization based recognition over sparse representation based recognition on several face image databases.

***Index Terms*—** quantization, visual information encoding/decoding, face recognition, identification

1. INTRODUCTION

Visual information classification is of great practical interest in many multimedia and security applications. Traditionally, human face recognition is considered to be a reference application for testing different recognition frameworks. The main reasons for the interest in automatic human face recognition systems are the wide range of real world practical applications such as identification, verification, posture/gesture recognition, social network linking and multi-modal interaction.

In the past, Nearest Neighbour (NN) [1] and Nearest Feature Subspace (NFS) [2] have been used for classification. NN classifies the query image by only using its Nearest Neighbour. It utilizes the local structure of the training data and is therefore easily affected by noise. NFS approximates the query image by using all the images belonging to an identical class, using the linear structure of the data. Class prediction is achieved by selecting that class of images that minimizes the

The research has been partially supported by SNF grant 1200020-146379 and by a grant from Switzerland through the Swiss Contribution to the enlarged European Union PSPB-125/2010. Correspondence: svolos@unige.ch

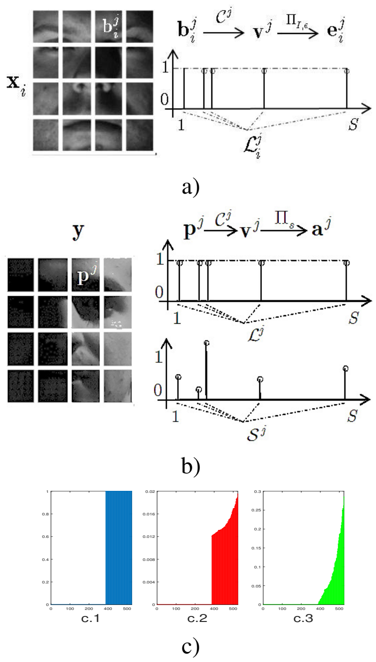

Fig. 1. a) Block encoding: Assignment of the block $\mathbf{b}_i^j$ to the nearest centroids from the set $\mathcal{C}^j$, $\Pi_{I,\epsilon}$ is an indicator function, $\mathbf{v}^j \in \Re^S$ represents the likelihood of the block $\mathbf{b}_i^j$ to the set of centroids $\mathcal{C}^j$, $\mathcal{L}_i^j$ is the list of centroid indexes. **b) Block decoding:** Assignment of block $\mathbf{p}^j$ to the nearest centroids from the set $\mathcal{C}^j$ using a constrained likelihood projection Π_s, $\mathbf{a}^j = \Pi_s(\mathbf{v}^j)$ is the projected vector, $\mathbf{v}^j \in \Re^S$ represents the likelihood of the block $\mathbf{p}^j$ to the set of centroids $\mathcal{C}^j$, $\mathcal{L}_j$ is the set of indexes for the centroids $\mathbf{c}_w^j$ and $\mathcal{S}_j$ is the set of corresponding coefficients obtained using $\Pi_s(\mathbf{v}^j)$, **c) Decoding types:** c.1 Hard decoding, c.2 ϵ-NN and c.3 $\frac{L_1}{L_2}$ norm constrained likelihood projection. Example of assignments coefficients that might be used by MVQ. Shown here are only those that correspond to the support (non-zero values) of the constrained likelihood projection. The sample is from the Yale B database.

reconstruction error. NFS might fail in the case that classes are highly correlated to each other. Certain aspects of these problems can be overcome by Sparse Representation based Classification (SRC) [3]. However, Qinfeng at al. [4] argue that the lack of sparsity in the data means that the compressive sensing approach cannot be guaranteed to recover the exact signal and therefore that sparse approximations may not deliver the desired robustness and performance. It has also been shown [5] that in some cases, the locality of the dictionary codewords is more essential than the sparsity. An extension of SRC, denoted as Weighted Sparse Representation based Classification (WSRC) [6] integrates the locality structure of the data into a sparse representation in a unified formulation and provides the best known recognition performance known so far for this family of methods.

Here we use a single decomposition of the Multilevel Vector Quantization (MVQ) approach for multiple levels of multi-resolution image representation presented by [7]. It should be pointed out that this method of visual information encoding/decoding has similarities with the *bag-of-features* (BoF) approach and Artificial Neural Networks (ANN). Since the core of the MVQ representation is based on vector quantization we will refer to this approach as vector quantization based recognition. Practically, we consider and compare two types of face recognition systems based on sparse representation and vector quantization.

This paper is organized as follows. Section 2 provides the basic problem formulation. In Section 4, we describe the proposed vector quantization method. The results of the computer simulations are presented in Section 4 and Section 5 concludes the paper.

Notation: We use capital bold letters to denote real valued matrices, $\mathbf{W} \in \Re^{N\text{x}KM}$, small bold letters to denote real valued vectors: $\mathbf{x} \in \Re^N$. The estimate of $\mathbf{x}$ is denoted as $\hat{\mathbf{x}}$. All vectors have finite length, explicitly defined where appropriate.

2. PROBLEM FORMULATION

The face recognition system consists of two stages: *Enrolment (coding)* and *Recognition (decoding)*.

At the enrolment stage, the facial photos from each subject are acquired and organized in the form of a codebook. We will assume that the recognition system should recognize *K* subjects. The photos of each subject are acquired under different imaging conditions such as lighting, expression, pose, etc., which will represent the variability of face features and serve as intra-class statistics. We will also assume that the frontal face images are aligned to the same scale, rotation and translation using common computer vision features.

We assume that every subject has M training samples. Each sample from subject k, $k \in \{1, \cdots, K\}$ is defined by a vector $\mathbf{x}_z \in \Re^N$, $z \in \{(1 + (k-1)M), \cdots, kM\}$ representing a concatenation of aligned image columns. Moreover we assume that the samples from all subjects are arranged into a codebook represented by a matrix:

$$\mathbf{W} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_{KM}] \in \Re^{N\text{x}KM}. \quad (1)$$

At the recognition stage, a probe or query $\mathbf{y} \in \Re^N$ is presented to the system. The system should recognize all subjects as accurate as possible based on $\mathbf{y}$ and the available $\mathbf{W}$. It is also assumed that $\mathbf{y}$ always corresponds to one of the subjects represented in the database. If it is not a case, a rejection option is integrated into the final decision.

3. MULTILEVEL VECTOR QUANTIZATION (MVQ) BASED RECOGNITION

In this section we present only the single level decomposition of the Multilevel Vector Quantization (MVQ) [7].

As a single level decomposition we refer to image partitioning into local block. The main idea behind the proposed method is to learn a codebook of centroids $\mathcal{C}^j = \{\mathbf{c}_1^j, \cdots, \mathbf{c}_S^j\}$ for each block j, $j \in \{1, \cdots, B\}$, where B is the number of blocks for a particular decomposition level, S stands for the number of centroids chosen to be the same for all block locations j.

The decomposition of images on local blocks is explained by the necessity to cope with the non stationary nature of distortions that are approximated by block-wise stationary ones using local block decompositions. The overall goal of the proposed method is to achieve a competitive classification accuracy together with an acceptable memory storage and complexity.

The MVQ based classification consists of three main steps: (a) codebook construction, (b) block encoding using the above codebook and (c) recognition based on the same codebook.

3.1. Codebook construction

Given the training data $\mathbf{x}_i$, $i \in \{1, \cdots, MK\}$ for all subjects K with M training samples each, each image is partitioned into B blocks of size $L = n \times n$ (where $n \in Z, n \leq \sqrt{N}$). Each block $\mathbf{b}_i^j \in \Re^L$ of a training image $\mathbf{x}_i$ is defined as $\mathbf{b}_i^j = \mathbf{M}^j \mathbf{x}_i$, $\mathbf{M}^j \in \Re^{L\text{x}N}$, $\mathbf{M}^j$ has ones only on the diagonal (equivalent to cropping the j-th block of pixels from the image). The codebook $\mathcal{C}^j$ for block j is generated using all the j image blocks from all the images $\mathbf{W}^j = \mathbf{M}^j\mathbf{W}$, $\mathbf{W}^j \in \Re^{L\text{x}KM}$, and applying Vector Quantization (k-means algorithm), resulting into a set of S centroids $\mathcal{C}^j = [\mathbf{c}_1^j, \cdots, \mathbf{c}_S^j]$.

3.2. Enrolment (coding)

Independently per any block location j, $j \in \{1, \cdots, B\}$ an encoding is performed. This encoding consists of two parts.

In the first part given a set of KM image blocks $\mathbf{W}^j = \mathbf{M}^j\mathbf{W} = [\mathbf{b}_1^j, \cdots \mathbf{b}_{KM}^j]$, coming from K subjects, where every subject has M samples, each block $\mathbf{b}_i^j$ is assigned to the nearest centroids using a k-NN or ϵ-NN strategy (bounded distance decoding), represented by a set (list):

$$\mathcal{L}_i^j = \left\{w : d(\mathbf{b}_i^j, \mathbf{c}_w^j) \leq \epsilon L, 1 \leq w \leq S\right\}, \tag{2}$$

where $d(.,.)$ is a distance metric (e.g. Euclidean distance), $\epsilon \geq 0$ and L is the block size. In equivalent vector form, the code vector $\mathbf{e}_i^j \in \Re^S$ for the above list is defined as:

$$e_i^j(w) = \mathbf{1}\left\{\exp\left(-\frac{d(\mathbf{c}_w^j, \mathbf{b}_i^j)}{\sigma}\right) \leq \epsilon L\right\}, \tag{3}$$

where $w \in \{1, \cdots, S\}$ and $\mathbf{1}\{.\}$ denotes the indicator function, σ is a parameter related to the second moment (e.g. in Gaussian distribution it represents variance). The value of σ is chosen such that $\sum_{w=1}^{S} \exp\left(-\frac{d(\mathbf{c}_w^j, \mathbf{b}_i^j)}{\sigma}\right) = 1$.

The encoding results in a generation of a total of KM code vectors $\mathbf{E}^j = [\mathbf{e}_1^j, \cdots, \mathbf{e}_{KM}^j] \in \{0,1\}^{S\text{x}KM}$ per block location j each with dimension S. This type of coding is shown in Figure 1 (a).

The second part consists of the subject encoding. This is performed by pooling the block code vectors of all the image blocks for a particular block position j that come from a particular subject k. This may be preformed by using average-pooling (AVGP):

$$\text{AVGP: } \mathbf{s}_k^j = \frac{1}{M}\sum_{z=1+(k-1)M}^{kM} \mathbf{e}_z^j, \tag{4}$$

where $k \in \{1, \cdots, K\}$. The main idea behind this particular form of pooling is to capture all centroids for a given block decomposition that might represent a subject under various observation distortions. In fact, if the observation model were stationary and known, the representative centroids could be computed analytically.

The final stage of encoding includes the generation of an inverted file look up table where each block $j \in \{1, \cdots, B\}$ has a set of centroids $w \in \{1, \cdots, S\}$ containing the indices of corresponding subjects $k \in \{1, \cdots, K\}$. This look up table is very sparse and efficient for memory storage.

3.3. Recognition (decoding)

The recognition consists of two parts: (a) block decoding and (b) decision fusion.

3.3.1. Block decoding

The goal here is to find a set (list) of similar block codes with the corresponding set of reliable coefficients.

Given an observation image $\mathbf{y}$ each block $\mathbf{p}^j = \mathbf{M}^j\mathbf{y}$, $\mathbf{p}^j \in \Re^L$, $j \in \{1, \cdots, B\}$ is assigned to the corresponding set of centroids $\mathcal{C}^j$ based on a constrained $\frac{L_1}{L_2}$ norm projection $\mathbf{a}^j = \Pi_s(\mathbf{v}^j)$ of the vector $\mathbf{v}^j$. In this assignment the vector $\mathbf{v}^j \in \Re^S$ represents the likelihood of the block $\mathbf{p}^j$ to the set of centroids $\mathcal{C}^j$ defined as:

$$\mathbf{v}^j = \left[\exp\left(-\frac{d(\mathbf{c}_1^j, \mathbf{p}^j)}{\sigma}\right), ..., \exp\left(-\frac{d(\mathbf{c}_S^j, \mathbf{p}^j)}{\sigma}\right)\right], \tag{5}$$

where $d(.,.)$ is a distance metric (e.g. Euclidean distance) and σ is the parameter related to the second moment (e.g. in Gaussian distribution it represents variance).The value of σ is chosen such that $\sum_{i=1}^{S} v^j(i) = \sum_{i=1}^{S} \exp\left(-\frac{d(\mathbf{c}_i^j, \mathbf{p}^j)}{\sigma}\right) = 1$. The projection vector $\mathbf{a}^j = \Pi_s(\mathbf{v}^j)$ is defined as a solution to the following constrained projection problem [8]:

$$\Pi_s : \hat{\mathbf{a}}^j = arg\min_{\mathbf{a}^j} \frac{1}{2}(\mathbf{v}^j - \mathbf{a}^j)^T(\mathbf{v}^j - \mathbf{a}^j),$$
$$\textit{subject to } \frac{\sqrt{N} - \frac{\|\mathbf{a}^j\|_1}{\|\mathbf{a}^j\|_2}}{\sqrt{N} - 1} = s, \tag{6}$$

where s is the predefined sparsity level. Though not presented in [8], one may prove that the same algorithm solves (6) for any $s \geq \frac{\sqrt{S} - \frac{\|\mathbf{v}^j\|_1}{\|\mathbf{v}^j\|_2}}{\sqrt{S} - 1}$ under weak assumptions about the distribution of the values of the elements of $\mathbf{v}^j$. However due to the space limitations we omit the proof.

In the decoding stage we might consider two sets. The first set is the list $\mathcal{L}_j$ of indexes for the centroids $\mathbf{c}_w^j$ that are most similar to the block $\mathbf{p}^j$ under the projection operator Π_s, and the second set $\mathcal{S}_j$ is the set of projected cluster block likelihood coefficients under the projection operator Π_s:

$$\mathcal{L}^j = \{w : a^j(w) > 0, 1 \leq w \leq S\},$$
$$\mathcal{S}^j = \{a^j(w) : a^j(w) > 0, 1 \leq w \leq S\}, \tag{7}$$

where $\mathbf{a}^j = \Pi_s(\mathbf{v}^j)$. This type of decoding is shown in (Figure 1 (b).

Given the vector $\mathbf{a}^j$ as an activation code and the subject block codes $\mathbf{s}_k^j$ the decoding similarity score is defined as:

$$t^j(k) = {\mathbf{s}_k^j}^T \mathbf{a}^j, 1 \leq k \leq K. \tag{8}$$

It is important to note that when only the list $\mathcal{L}^j$ is used to construct the activation code, the method is considered to be *hard decoding* (i.e., $a^j(w) = 1, w \in \mathcal{L}^j$ in (7). On the other hand when the two sets $\mathcal{L}^j$ and $\mathcal{S}^j$ are used to construct the activation code, the method is considered to be *soft decoding*. In the case of soft decoding, one may expect that the reliable centroids will obtain weights from the set $\mathcal{S}^j$ closer to 1 and non-reliable closer to 0. It is also remarkable that in the case of reliable decoding, for a particular high sparisty level s the number of elements of the set $\mathcal{S}^j$ will be significantly smaller then S indicating that the reliable centroid(s) is(are) found.

Otherwise, all elements of this set will have identical weights. Therefore, one can use the notion of sparsity to estimate the reliability of the produced estimate (as an example in Figure 1 (c) we show the distribution of the values of the elements in the hard and soft assignment vectors).

3.3.2. Decisions fusion

Basic fusion: The final decision using a particular decomposition is produced as a result of the most likely subject $k \in \{1, \cdots, K\}$ selection that obtains the majority of votes:

$$\hat{i} = \max_{1 \le k \le K} \sum_{j=1}^{B} t^j(k) = \max_{1 \le k \le K} \sum_{j=1}^{B} {\mathbf{s}_k^j}^T \mathbf{a}^j. \quad (9)$$

Weighted fusion: Given $\mathbf{t}^j$, it's sparsity level is denoted as $h(j) = \frac{\sqrt{K} - \frac{\|\mathbf{t}^j\|_1}{\|\mathbf{t}^j\|_2}}{\sqrt{K}-1}$, $j \in \{1, \cdots, S\}$ and the vector of weight coefficients is defined as $\mathbf{h}_s = \Pi_s(\mathbf{h})$, where Π_s is the projection defined by equation (6). This fusion as a result produces the most likely subject k selection that obtains the $h_s(j)$ weighted majority of the votes:

$$\hat{i} = \max_{1 \le k \le K} \sum_{j=1}^{B} h_s(j) \mathbf{t}^j(k). \quad (10)$$

4. COMPUTER SIMULATIONS

In this section we present the results of the computer simulation. The computer simulation is performed to compare the recognition rate of WSRC versus MVQ in three different setups under: *varying dimensionality, random pixel corruption and continuous occlusion*.

The computer simulation is carried out on 4 publicly available face datasets, namely: Extended Yale B [9], AR [10] PUT [11] and FERET [12]. The Extended Yale B consists of 2414 frontal face images of 38 subjects captured under various laboratory-controlled *extreme lighting variability*. All the images from this database are cropped and normalized to 192x168 pixels. The AR database consists of over $4,000$ frontal images for 126 individuals. For each individual, 26 pictures were taken in two separate sessions. These images include a variety of facial variations, including illumination change, *expressions variability*, and facial disguises. In the experiment here, we chose a subset of the data set consisting of 100 subjects. For each subject, 14 images with only illumination change and expressions were selected. The images are cropped and normalized to 165x120 pixels. The PUT database consists of hi-resolution images of 100 people. Images were taken in controlled conditions under various *pose variation*. In our set up, we use a total of 2200 cropped and normalized to 178x178 pixels. The FERET database consists of 13,539 facial images corresponding to 1,565 subjects, who are diverse across ethnicity, gender, and age. In our experiments, we used two subsets FERET-1 and FERET-2 from the FERET database. More precisely we used frontal face images from the sets Fa and Fb and in total used 546 images downscaled to resolution 128x128.

Recognition rates (%)								
	Yale B				**AR**			
Dimension	30	42	120	504	30	54	130	540
WSRC	**79.3**	**88.2**	93.8	96.6	58.9	70.1	83	92.8
MVQ	53.1	80.1	**95.5**	**98.3**	**62.9**	**80.8**	**94.3**	**98.1**
	PUT				**FERET**			
Dimension	36	64	121	484	25	49	100	441
WSRC	80.1	90	93.1	96.2	41.7	45.7	58	79.1
MVQ	**92.9**	**96.8**	**98.1**	**98.9**	**58.8**	**79.7**	**88.3**	**91**

Table 1. WSRC versus MVQ: Recognition results on Yale B, AR, PUT and FERET databases using different image dimensions

For all images in the above datasets, the facial part of each image was manually cropped, aligned according to eyes positions.

In all of the computer simulations the face images are converted to gray scale. For all of the images we use raw, basic, elementary image pixel values (block of image pixel values) as features. To be unbiased in our validation of the results we use 5-fold cross validation, where for single validation for each subject, half of the images are selected at random for training and the remainder for testing.

In all the experiments the MVQ model uses trained codebooks that consists of a set of S centroids, the number of the codebooks $\mathcal{C}^j$ is equivalent to the number of block locations B. For any block location j the codebook $\mathcal{C}^j = \{\mathbf{c}_1^j, \cdots, \mathbf{c}_S^j\}$ is learned with the k-means algorithm. The number of centroids in every codebook is equivalent to the half of the available training data and the block size $L = 9$. In the block encoding part the sparsness coefficient s in (6) is set to high value . In the decision fusion part we use the rule (10) and the sparsness coefficient is set to low value.

As an implementation of WSRC here we use cvx [13] with Mosek 6.9 solver and WSRC parameters as specified in the original paper [6].

We test and compare the recognition rate of the WSRC and MVQ methods under varying dimensionality on all of the four databases. In the case of random noise corruption and continuous occlusion the robustness comparison is made using the Extended Yale B face database.

4.1. Recognition under varying dimensionality

In this experiment the used downsample ratios per database Yeale B are $1/32, 1/24, 1/16$ and $1/8$; for the databases AR and FERET are $1/24, 1/18, 1/12$ and $1/6$; and for database PUT are $1/28, 1/21, 1/16$ and $1/8$. The results are shown on Table 1. As we may see from this results the MVQ method

Recognition rate under random corruptions (%)										
Corruption (%)	0	10	20	30	40	50	60	70	80	90
WSRC	**100**	**100**	**100**	98	95	79	57	33	18	6
MVQ	**100**	96	86	67	46	32	23	13	8	5
MVQ*	**100**	**100**	**99**	**99**	**97**	**90**	**79**	**52**	**23**	**7**
Recognition rate under continuous occlusion (%)										
Occluded (%)	0	10	20	30	40	50				
WSRC	**100**	**100**	96	89	78	61				
MVQ	**100**	99	**98**	**97**	**94**	**89**				

Table 2. WSRC versus MVQ: Recognition rate under random corruptions and varying level of continuous occlusion, in MVQ* the block size is set to $L = 225$.

has consistently higher recognition rates on all databases per all of the chosen dimensions except for the two smallest dimensions at the Yale B database.

4.2. Recognition despite random pixel corruption and continuous occlusion

Here the setup is equivalent to the one defined in [3]. he various level of random noise, from 0 percent to 90 percent, are simulated by corrupting a percentage of randomly chosen pixels from each of the test images, replacing their values with independent and identically distributed samples from a uniform distribution. The various levels of contiguous occlusion, from 0 percent to 50 percent, are simulated by replacing a randomly located square block of each test image with an unrelated image. The results are shown in Table 2. As we may see from these results the MVQ method has consistently higher recognition rates under random noise corruption however at a cost of using bigger image blocks. This may be explained by the fact that the bigger blocks are less effected by the uniform noise. In the later case the blocks that come from the non-occluded regions are crucial for reliable and accurate results.

We may thus conclude that by using a simple, local decomposition representation with proper reliability estimates such as MVQ presented here, one may achieve high recognition accuracy.

5. CONCLUSIONS

In this paper we considered the face recognition problem from both machine learning and information coding perspective, adopting an alternative way of visual information encoding and decoding. Our model for recognition is based on *multi-level vector quantization* (MVQ), conceptually similar to BoF and CNN. The results from the computer simulations confirm that the MVQ based recognition model has a superior accuracy over weighted sparse representation base recognition [6] on several face image databases.

REFERENCES

[1] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.

[2] S. Shan, W. Gao, and D. Zhao, "Face recognition based on face-specific subspace," *International journal of imaging systems and technology*, vol. 13, no. 1, pp. 23–32, 2003.

[3] J. Wright, A. Y Yang, A. Ganesh, S. S Sastry, and Yi Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[4] Qinfeng S., A. Eriksson, A. van den Hengel, and Chunhua S., "Is face recognition really a compressive sensing problem?," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 553–560, 2011.

[5] A. Coates and A. Y Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 921–928.

[6] Can-Yi Lu, Hai Min, Jie Gui, Lin Zhu, and Ying-Ke Lei, "Face recognition via weighted sparse representation," *Journal of Visual Communication and Image Representation*, vol. 24, no. 2, pp. 111–116, 2013.

[7] Kostadinov D., Voloshynovskiy S., and Diephuis M., "Visual information encoding for face recognition: sparse coding vs vector quantization," in *4th Joint WIC IEEE Symposium on Information Theory and Signal Processing in the Benelux*, Eindhoven, Netherlands, 05/2014 2014, vol. 35.

[8] P. O. Hoyer and P. Dayan, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[9] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 643–660, 2001.

[10] A. Martínez and R. Benavente, "The ar face database," Tech. Rep. 24, Computer Vision Center, Jun 1998.

[11] A. Schmidt A. Kasinski, A. Florek, "The put database," Tech. Rep. 3, CIE Bioinformatics, 2008.

[12] P. Jonathon Phillips, Hyeonjoon M., Syed A. R., and P. J. Rauss, "The feret evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[13] Inc. CVX Research, "Cvx, matlab software for disciplined convex programming, version 2.0," aug 2012.