

# Model-based reconstruction for illumination variation in face images

B.J.Boom, L.J.Spreeuwers, and R.N.J.Veldhuis

Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente  
P.O.Box 217 7500 AE Enschede, The Netherlands

b.j.boom@ewi.utwente.nl

## Abstract

*We propose a novel method to correct for arbitrary illumination variation in the face images. The main purpose is to improve recognition results of face images taken under uncontrolled illumination conditions. We correct the illumination variation in the face images using a face shape model, which allows us to estimate the face shape in the face image. Using this face shape, we can reconstruct a face image under frontal illumination. These reconstructed images improve the results in face identification. We experimented both with face images acquired under different controlled illumination conditions in a laboratory and under uncontrolled illumination conditions.*

## 1. Introduction

In video surveillance applications, face recognition has been a difficult problem due to variations in the face, like illumination, pose and facial expressions. In this paper, we introduce a method for correcting for the illumination variations. Correcting for these variations is necessary because they are often larger than the variation due to changes of the person's identity. We propose a method that is able to correct the illumination in a single image.

Several approaches have already been proposed in literature, that make face images invariant for illumination. These approaches can be divided into two categories. The first category works by applying a preprocessing step to the images, like Histogram Equalization [9] or Local Binary Patterns [5]. These methods are direct and simple but often lack a theoretical explanation. The second category works using physical models of the illumination mechanism and its interactions with the object surface. The Illumination Cone [4] falls in this category, which estimates the shape from at least three images under different illumination conditions. An other method is the Quotient Image [10] which estimates illumination of a single image allowing them to compute a quotient image. This method does not model shadows and reflection. The method proposed in [11] can

deal with shadows and reflections by adding an error term. In the 3D morphable model described in [2], the Phong model is applied to simulate the reflection, also shadows are properly modeled by the 3D shape of the face. In our method, we choose to model shadows and reflection by adding an error term as in [11], making our method computationally efficient, but we use a shape model derived from 3D range maps, allowing us to estimate the shape from a single image.

Our paper is organized as follows, in Section 2 we describe our method to correct illumination variation in face images. In Section 3 we describe the experiments and the results of the face recognition algorithm on the reconstructed face images. In Section 4 we present some conclusions.

## 2. Method

### 2.1. Lambertian model

In order to solve the illumination problem in face recognition for a single image, we begin with some simple assumptions. Our first assumption is that we have a single light source at infinite distance, making the problem easier and computationally tractable. The direction and intensity of the light source is not known to us. Recovering an unknown shape for a 2D image without further knowledge is impossible, because many 3D shapes result in the same 2D image. For this reason we use a 3D face shape model which allows us to estimate the face shape in the face image. In this paper, we assume that the illumination of faces behaves to a large extent according to the Lambertian equation. We introduce an error term to model shadows and reflections which are normally not included in the Lambertian equation [11]. This gives the following Lambertian equation:

$$b(x) = c(x)\mathbf{n}(x)^T \mathbf{s}i + e(x; \mathbf{s}) \quad (1)$$

$x$  is the pixel position and  $b \in \mathcal{R}$  is the pixel intensity in the image. The pixel intensity is determined by the dot product of the shape and the illumination. The surface normals  $\mathbf{n} \in \mathcal{R}^3$ , which define the direction of the reflection, and the albedo of the surface given by  $c \in \mathcal{R}$  together are the shape

$\mathbf{h}(x) = c(x)\mathbf{n}(x)^T$ . The direction of the light, which is a normalized vector given by  $\mathbf{s} \in \mathcal{R}^3$ , and the intensity of the light, which is given by  $i \in \mathcal{R}$ , describe the light conditions  $\mathbf{v} = si$ . The error term  $e \in \mathcal{R}$  that we introduce allows us to handle reflection and shadows which are not modeled in the Lambertian equation. Instead of writing these term for every pixel position  $x$ , we can also vectorize the image giving us the following equation:

$$\mathbf{b} = H\mathbf{v} + \mathbf{e}(\mathbf{s}) \quad (2)$$

In our case we have  $M$  pixel positions, which gives us the vectorized face image  $\mathbf{b} \in \mathcal{R}^{M \times 1}$ , a matrix which contains the face shape  $H \in \mathcal{R}^{M \times 3}$  and the error term  $\mathbf{e}(\mathbf{s}) \in \mathcal{R}^{M \times 1}$ .

## 2.2. Overview of our correction method

In this paper, we want to correct the illumination in a single image, where we only have the pixel intensities  $b(x)$  of the Lambertian equation. The other terms, like the shape  $\mathbf{h}(x)$  and light  $\mathbf{v}$  have to be estimated by our method. Our method uses a model of the face shape, which allows us to calculate the parameters of the face shape given that we know the light parameters. Because the lighting is unknown, we search for the light parameters which give the optimal face shape parameters. For different light directions, we estimate a light intensity and calculate the face shape. We evaluate all the face shapes calculated under different light directions. Using kernel regression, we are then able to determine the final face shape from the different face shapes. The different steps of our method are given below in pseudo-code:

- Learning a model of the face shape (offline)
- For different light direction  $\mathbf{s}_j$ 
  - Calculate a shadow and reflection term  $e_j(x; \mathbf{s}_j)$
  - Estimate the light intensity  $i_j$  which gives us light source  $\mathbf{v}_j$
  - Fit the face shape model to the face image which gives us the shape  $\mathbf{h}_j(x)$
  - Evaluate the shape  $\mathbf{h}_j(x)$  which gives us a distance measure  $d_j$
- Calculate the final shape  $\mathbf{h}(x)$  using kernel regression
- Refine the albedo of the surface  $c(x)$  to obtain more details

Throughout our paper,  $j$  is a index for the different light direction, where we calculate for  $J$  light directions error terms  $e_j(x; \mathbf{s}_j)$ , light intensities  $i_j$ , face shapes  $\mathbf{h}_j(x)$  and distance measures  $d_j$ . Using the different shapes and distances, we can obtain one final shape which we refine. With this final shape, we can calculate a face image under frontal illumination conditions, using the following equation:

$$b_{\text{frontal}}(x) = \mathbf{h}(x)^T \mathbf{v}_{\text{frontal}} \quad (3)$$



Figure 1. The mean shape and first five deviation of the shape model, on the face shape we draw the face image under frontal illumination

In the next sections, we will discuss the different steps described in our pseudo code in more detail.

## 2.3. Learning the Face Shape Model

A set of face images and 3D range maps provide us with the means to calculate the face shape  $H$  for each face image (See Section 3.2). Because we have multiple face shapes we can determine a mean shape  $\bar{H}$  and variations from this mean shape. To calculate the variations, we vectorized all the  $L$  obtained shapes into the data matrix  $X \in \mathcal{R}^{3M \times L}$  from which we compute the covariance matrix  $\Sigma$ . Using Principal Component Analysis (PCA), we can decompose  $\Sigma$  as  $\Sigma = \Phi\Lambda\Phi^T$ , where the columns of  $\Phi$  are the eigenvectors and the matrix  $\Lambda$  contains the eigenvalues on the diagonal. The columns of  $\Phi \in \mathcal{R}^{3M \times K}$ , are converted to  $M \times 3$  matrices, which we denote by  $T_k \in \mathcal{R}^{M \times 3}$ . These matrices  $\{T_k\}_{k=1}^K$  contain the most important variations of the face shape. In Figure 1, we calculated the surface from the shape and on this surface we have drawn a reconstructed image under frontal illumination. The left image on top is the mean shape and the other images from left to right are the first 5 most important variations in the shape model.

## 2.4. Shadow and Reflection Term

In this paper we use the Lambertian equation as basis, which is unable to deal with shadows and reflection. For this reason, we added the error term  $e(x; \mathbf{s})$  in the equation 1. We assume that the error term depends on the light direction  $\mathbf{s}$ , which is the determining factor for shadows and reflections. We use a 2D face database with labelled illumination conditions to learn the error term for each light direction. Using the kernel regression method described in [11], we can calculate for a light direction  $\mathbf{s}_j$  (in Section 2.2) the mean error  $\mu_e(x, \mathbf{s}_j)$  and the variance  $\sigma_e^2(x, \mathbf{s}_j)$ . For the error term in equation 1, we take  $e_j(x; \mathbf{s}_j) = \mu_e(x, \mathbf{s}_j)$ . The variance  $\sigma_e^2(x, \mathbf{s}_j)$  will be used later in the Section 2.9. Although we use a face-independent mean term for the shadows and reflections, this estimation is better than ignoring the error term.

## 2.5. Light Intensity

We want to estimate a face shape using the face image and our face shape model, where we know the light direction  $\mathbf{s}_j$  and the error term  $\mathbf{e}_j(\mathbf{s}_j)$ . To estimate this shape we have to know the light intensity  $i$  which allows us to calculate the light conditions  $\mathbf{v} = si$ . To calculate the intensity, we replace the unknown shape  $H$  with the mean shape  $\bar{H}$ . Using a linear solver, we are able to solve the following equation, where the light intensity  $i_j$  is the unknown:

$$i_j = \arg \min_{i_j} \|(\bar{H}\mathbf{s}_j)i_j - (\mathbf{b} - \mathbf{e}_j(\mathbf{s}_j))\|^2 \quad (4)$$

Because this is a overcomplete system, we can use the mean face shape  $\bar{H}$  to estimate the light intensity  $i_j$ , which still gives a very accurate estimation. However, this might normalize the different intensities in the skin color of different persons.

## 2.6. Estimation of the Face Shape

In the previous sections, we calculated for a light direction  $\mathbf{s}_j$ , the error term  $\mathbf{e}_j(\mathbf{s}_j)$  and light condition  $\mathbf{v}_j$ . Using these terms we are now able to obtain the face shape  $H_j$ . In this section, we will calculate the shape using the face shape model obtained in Section 2.3. Using the face shape model, we can replace  $H_j$  with:

$$H_j = \bar{H} + \sum_{k=1}^K T_k y_{j,k} \quad (5)$$

This is the mean shape and the  $K$  most important variations. Our approach has much in common with 3D morphable models [2], however we apply it solely for shape recovery. We can now rewrite the Lambertian equation as follows 1:

$$\bar{H}\mathbf{v}_j + \sum_{k=1}^K T_k \mathbf{v}_j y_{j,k} = \mathbf{b} - \mathbf{e}_j(\mathbf{s}_j) \quad (6)$$

$$\sum_{k=1}^K T_k \mathbf{v}_j y_{j,k} = \mathbf{b} - \mathbf{e}_j(\mathbf{s}_j) - \bar{H}\mathbf{v}_j \quad (7)$$

Instead of calculating the shape  $H_j$ , we are now able to calculate the variations  $\mathbf{y}_j \in \mathcal{R}^K$  of the shape using a linear solver. In this case, we write  $T_k \mathbf{v}_j = A_k$  and  $\mathbf{b} - \mathbf{e}_j(\mathbf{s}_j) - \bar{H}\mathbf{v}_j = \mathbf{c}$  giving us the following linear solvable equation:

$$\mathbf{y}_j = \arg \min_{\mathbf{y}_j} \|\mathbf{A}\mathbf{y}_j - \mathbf{c}\|^2 \quad (8)$$

To calculate the shape  $H_j$  from the parameters  $\mathbf{y}_j$  obtained in Equation 8, we use Equation 5. In this case, we are able to estimate the shape given the direction of the illumination  $\mathbf{s}_j$ . In the next section, we explain how we evaluated the obtained face shape.

## 2.7. Evaluation of the Face Shape

To evaluate the face shape we made two observations: First, the found variations will be small when the light direction is similar to the light direction in the face image. Second, when the light directions are similar, the found variations of the face shape create a reconstructed image which matches the input image. These criteria are similar to the criteria used in the active shape models [3]. Although in their case, they can compare their output directly with the image, we have to convert our found shape  $H_j$  along with the light direction  $\mathbf{s}_j$  to a reconstructed image  $\mathbf{b}_j$ , which is calculated as following:

$$\mathbf{b}_j = H_j \mathbf{v}_j + \mathbf{e}_j(\mathbf{s}_j) \quad (9)$$

Because we calculate for different light directions  $\mathbf{s}_j$  new reconstructed images  $\mathbf{b}_j$ , we can compare these images with the original image  $\mathbf{b}$ . We do this by calculating the sum of the square differences between both images as follows:

$$\epsilon^2 = (\mathbf{b} - \mathbf{b}_j)^T (\mathbf{b} - \mathbf{b}_j) \quad (10)$$

We know that the reconstructed image  $\mathbf{b}_j$  contains the variations  $\mathbf{y}_j$  to the face shape model, because  $\mathbf{v}_j$  and  $\mathbf{e}_j(\mathbf{s}_j)$  are constants. Our face shape model has only  $K$  variations, making it impossible to explain all deviations in the face images. In [3], the distance measure how well the model fits is given by:

$$d_j = \sum_{k=1}^K \frac{y_k^2}{\lambda_k} + \frac{\epsilon^2}{\rho} \quad (11)$$

We use the estimation of [7] for  $\rho = \frac{1}{3M-K} \sum_{k=K+1}^{3M} \lambda_k$ . Using this distance measure, we can easily evaluate the quality of the found shape for a certain light direction.

## 2.8. Calculate final shape using kernel regression

In Section 2.6, we calculated the face shape parameters  $\{\mathbf{y}_j\}_{j=1}^J$  for different light directions  $\mathbf{s}_j$ . In Section 2.7, we evaluate the distance measures  $\{d_j\}_{j=1}^J$ , which determine the quality of the different face shape parameters. The light directions  $\mathbf{s}_j$  are the same as the light directions used in the face database with labelled illumination conditions. The main reason that we use the same light directions is because these light directions cover a complete grid of light directions. This is ideal for applying kernel regression. Using the obtained face shape parameters  $\{\mathbf{y}_j\}_{j=1}^J$  and the distances  $\{d_j\}_{j=1}^J$  our regression method can be seen in the following equations:

$$\mathbf{y} = \sum_{j=1}^J w_j \mathbf{y}_j / \left( \sum_{j=1}^J w_j \right) \quad (12)$$

$$w_j = \exp\left[-\frac{1}{2}(d_j/\sigma)^2\right] \quad (13)$$

In the above equation,  $\sigma$  is determined so that 5 percent of the distances lie within  $1 \times \sigma$ . The final shape parameters obtained from  $\mathbf{y}$  give us the final shape  $H$  using Equation 5. We can calculate the light conditions  $\mathbf{v}$  using a linear solver, which allows us to determine the final light conditions.

## 2.9. Refinement

The obtained shape can be divided into two parts, the first part are the surface normals  $\mathbf{n}(x)$ , the second part is the albedo of the surface denoted by  $c(x)$ . After estimating the final shape  $H$ , we observed that the reconstructed frontal illuminated image from shapes does not contain all details present in the original face image. Partially, this phenomenon can be explained by the fact that we do a dimension reduction. Another part can be explained by the fact that the kernel regression performs an interpolation. To recover these details, we recalculate only the albedo of the surface  $c(x)$ . Because the albedo of the surface contains most of the details, while the surface normals contain the larger structures of the face shape. To calculate the albedo  $c(x)$  we use a MAP estimate give by the following equation:

$$c(x)_{MAP} = \arg \max_c P(b(x)|c(x))P(c(x)) \quad (14)$$

As can be seen from Equation 14, we estimate the albedo for every pixel. For clarity, we will drop the " $(x)$ " in the following equations and replace the surface normals and final light condition  $\mathbf{n}(x)^T \mathbf{s}i$  with the constant  $q$ . The albedo of the surface  $c(x)$  can be estimated by the following equation, where we assume two gaussian distributions:

$$c_{MAP} = \arg \max_c \mathcal{N}(cq + \mu_e(\mathbf{s}), \sigma_e^2(\mathbf{s})) \times \mathcal{N}(\mu_c, \sigma_c^2) \quad (15)$$

$$\arg \min_c L = \left( \frac{q - cq - \mu_e(\mathbf{s})}{\sigma_e(\mathbf{s})} \right)^2 + \left( \frac{c - \mu_c}{\sigma_c} \right)^2 \quad (16)$$

The mean and variance of the error term are calculated with kernel regression, which is described in Section 2.4. The log probabilities  $L$  are given in Equation 16. We find the minimum by taking the derivative and make it equal to zero. The new albedo  $c_{MAP}$  contains more details than the albedo obtained using the PCA model. We have observed that the details are very important in the face recognition processes.

## 3. Experiments and Results

The purpose of the illumination correction is to improve the recognition rate of the face classifier. To evaluate this we performed two experiments to test if our correction algorithm indeed improves the recognition results. The first experiment is done on the Yale B databases to see if our algorithm can deal with different illumination conditions. The second experiment is performed on the FRGCv1 database where we use controlled face images for enrollment and the uncontrolled face images for the classification.

### 3.1. Face databases for Training

In the Section 2, we described our method which uses two different face databases for training. In this paper we use publicly available face databases. For the shadows and reflections, we used the Yale B [4] and extended Yale B [6] database (for short just the Yale B databases), these databases contain face images illuminated under different labelled light directions. To make a shape model, we needed a face database which contains both images and 3D range maps. The Face Recognition Grand Challenge (FRGC) [8] database is a publicly available database with a subset which contains 3D range maps. To make the shape model we used a subset of the entire 3D FRGC database namely the Spring 2003 3D range maps and face images. These face images contain almost frontal illumination and no shadows, making this subset of the database ideal for the calculation of the shape model. We performed some simple spike removal and filling techniques to obtain better surfaces from which we calculated the surface normals. Using these surface normals we are able to derive the shape model.

### 3.2. Determine albedo of the Shape

From the range maps, we are able to calculate the surface normals  $\mathbf{n}(x)$  for each pixel in the image. In this case, we obtained for every pixel the image intensity  $b(x)$  and the surface normals  $\mathbf{n}(x)$ . Looking at the Lambertian equation 1, we need for the shape  $h(x)$  also the albedo of the surface  $c(x)$ . Because the face images and 3D range maps were taken under almost frontal lighting, we decided to ignore the  $e(x)$  term. In this case we only have to obtain the light conditions, to be able to calculate the albedo of the surface  $c(x)$ . To estimate the light conditions  $\mathbf{v}$  we determined the mean albedo of the surface  $\mu_c(x)$  from the Yale B databases, which allows us to estimate  $\mathbf{v}$  as follows:

$$\begin{aligned} \mathbf{g}(x) &= \mu_c(x)\mathbf{n}(x) \\ \mathbf{v} &= \arg \min_{\mathbf{v}} \|\mathbf{G}\mathbf{v} - \mathbf{b}\|^2 \end{aligned} \quad (17)$$

In Equation 17, we first calculate the temporary shape  $\mathbf{g}(x)$  to estimate the light conditions  $\mathbf{v}$ . We represent the temporary shape in the matrix  $G \in \mathcal{R}^{M \times 3}$ . Using a linear solver we are able to calculate  $\mathbf{v}$ , see Equation 17. Using the light conditions  $\mathbf{v}$  we can calculate the albedo of the surface with the following equation:

$$c(x) = \frac{b(x)}{\mathbf{n}(x)^T \mathbf{v}} \quad (18)$$

We calculate for every pixel the albedo of the surface  $c(x)$ , sometimes these values become very large because the surface normals are perpendicular with the light source. In those cases we use a filling algorithm to correct for these mistakes.



Figure 2. Face images from the Yale B database, the first row contains uncorrected images, the second row contains the correction of [11] and the last row is corrected using our method.

### 3.3. Face Recognition

The illumination correction is a preprocessing step to improve the face recognition. For this reason, we performed a recognition experiment on two face databases to see if the correction indeed helps improving the recognition results. For the face recognition, we performed a feature reduction by subsequently using a PCA [12] and LDA [1] transformation to the corrected face images, using 200 PCA and 50 LDA components. After feature reduction, we use the likelihood ratio, described in [13] to obtain the similarity scores. In the next sections, we discuss how this classifier is used to obtain the results on the different databases.

### 3.4. Yale B database

The Yale B databases are in our case used to train the illumination model. These databases also allow us to evaluate our algorithm under different illumination conditions. Because we trained our illumination model on the Yale B databases, we performed a leave one person out experiment, with all the face images with the azimuth and elevation angle below the  $\pm 90$  degrees. In Figure 2, we show the original and reconstructed face images of a person in the Yale B database, where we used both our method and the method describe in [11] for correction. In the case of big shadows, like the eyes in the most right image in Figure 2, the method in [11] corrects by filling in the mean face. Our method however is bound to the shape model which also has to explain dependencies between pixels, which allows us to correct more user-specific

We used all the corrected face images obtained from the Yale B databases for a recognition experiment. We trained on thirty persons and used the face images of the remaining eight persons for enrollment and testing. We compared each image with the other images taken under all the different illumination conditions, where we left out the images which are taken under similar illumination conditions. We repeated this experiment until we obtained the similarity scores for every person with all the other persons in the

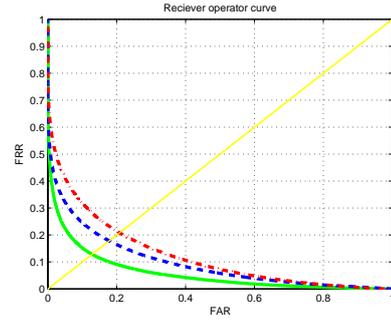


Figure 3. The ROC of the face recognition experiment on the Yale B databases, where the solid line is our correction method, the dashed line is the method describe in [11] and the dotted-dashed line is for the uncorrected images



Figure 4. Face images from the FRGCv1 database, the first row contains uncorrected images, the second row contains the corrected images, with for both persons a controlled and uncontrolled images.

database. The Receiver operating characteristic (ROC) of this experiment is shown in Figure 3. For this experiment, we clearly see that our method (EER: 12.83%) performs better than the method in [11] (EER: 17.84%) and the uncorrected images (EER: 20.82%).

### 3.5. FRGCv1 database

The main purpose of our method is to solve the surveillance problem where we have face images taken under controlled conditions but we also want to recognize these persons under uncontrolled conditions. In our case, the main focus is the illumination correction of these face images. The Face Recognition Grand Challenge version 1 contains frontal face images taken under both controlled and uncontrolled conditions which allows us to setup an experiment using this database. We corrected the illumination effects on all images in the FRGCv1 database, both the controlled and uncontrolled conditions. Examples of some reconstructed face images from the FRGCv1 database are shown in Figure 4.

For our recognition experiment, we randomly divided the uncontrolled and controlled face images into two parts, each

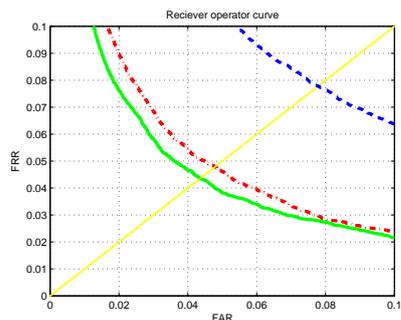


Figure 5. The ROC of the face recognition experiment on the FR-CGv1 database, where the solid line is our correction method, the dashed line is the method describe in [11] and the dotted-dashed line are the uncorrected images

containing approximately half of the face images. We used the first halves of both sets to train our face classifier, the second half of the controlled images are used for the enrollment of the one user template for every user and the second half of the uncontrolled images are used as probe images. We repeated this experiment 20 times randomly splitting the database to remove statistical fluctuations. The ROC curves of our experiment are shown in Figure 5, where our method obtained a EER of 4.35 %, while using uncorrected images we obtain a EER of 4.80 %. We also show the results of [11], but their method is not able to deal with different light intensities because it has been trained on the Yale B databases which only contains one intensity. Although our method improves the recognition rates in this experiment, the improvement is smaller than the one reported on the Yale B database. A reason is that the FRGC database contains other challenges like small poses, expressions, motion blur, while the Yale B database only focusses on illumination problems.

#### 4. Conclusion

In this paper, we propose a novel approach to correct face images for unknown illumination conditions. By fitting a face shape model under different light directions on the face images, we are able to estimate the face shape from which we can reconstruct a face image under frontal illumination. To test if these reconstructed face images improve the recognition rates we setup two experiments. In our first experiment, we achieve better recognition rates on the reconstructed face images acquired in a laboratory under different kinds of light conditions. The second experiment shows our method also improves the recognition results under uncontrolled illumination, making the algorithm suitable for surveillance applications.

#### References

- [1] P. N. Belhumeur, J. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *ECCV 2*, 1996.
- [2] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. pages 187–194, 1999.
- [3] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. In *IPMI '93: Proceedings of the 13th International Conference on Information Processing in Medical Imaging*, pages 33–47, London, UK, 1993. Springer-Verlag.
- [4] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.
- [5] G. Heusch, Y. Rodriguez, and S. Marcel. Local binary patterns as an image preprocessing for face authentication. *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, pages 6 pp.–, 10–12 April 2006.
- [6] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 27(5):684–698, 2005.
- [7] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):696–710, 1997.
- [8] NIST. Frgc face db. <http://www.frvt.org/FRGC/>.
- [9] S. Shan, W. Gao, B. Cao, and D. Zhao. Illumination normalization for robust face recognition against varying lighting conditions. *Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop on*, pages 157–164, 17 Oct. 2003.
- [10] A. Shashua and T. Riklin-Raviv. The quotient image: class-based re-rendering and recognition with varying illuminations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(2):129–139, Feb 2001.
- [11] T. Sim and T. Kanade. Combining models and exemplars for face recognition: An illuminating example. In *Proc. CVPR Workshop on Models versus Exemplars in Computer Vision*, december 2001.
- [12] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, pages 71–86, 1991.
- [13] R. Veldhuis, A. Bazen, W. Booij, and A. Hendrikse. Hand-geometry recognition based on contour parameters. In *Proceedings of SPIE Biometric Technology for Human Identification II*, pages 344–353, Orlando, FL, USA, March 2005.