Automatic Facial Makeup Detection with Application in Face Recognition[*]

Cunjian Chen
Computer Science and Electrical Engineering
West Virginia University, Morgantown, USA
cchen10@csee.wvu.edu

Antitza Dantcheva, Arun Ross
Computer Science and Engineering
Michigan State University, East Lansing, USA
antitza@msu.edu, rossarun@cse.msu.edu

Abstract

Facial makeup has the ability to alter the appearance of a person. Such an alteration can degrade the accuracy of automated face recognition systems, as well as that of methods estimating age and beauty from faces. In this work, we design a method to automatically detect the presence of makeup in face images. The proposed algorithm extracts a feature vector that captures the shape, texture and color characteristics of the input face, and employs a classifier to determine the presence or absence of makeup. Besides extracting features from the entire face, the algorithm also considers portions of the face pertaining to the left eye, right eye, and mouth. Experiments on two datasets consisting of 151 subjects (600 images) and 125 subjects (154 images), respectively, suggest that makeup detection rates of up to 93.5% (at a false positive rate of 1%) can be obtained using the proposed approach. Further, an adaptive pre-processing scheme that exploits knowledge of the presence or absence of facial makeup to improve the matching accuracy of a face matcher is presented.

1. Introduction

The matching accuracy of automated face recognition systems has significantly improved over the past decade [8]. Indeed, challenges related to variations in pose, illumination and expression (PIE) have been identified and addressed by advanced algorithms that allow for unconstrained face recognition in diverse applications [8]. In spite of these advancements, there are still several factors that continue to challenge the performance of face recognition systems. These include factors related to aging [7], plastic surgery [3], and spoofing [10]. In a recent paper, Dantcheva et al. [5] demonstrated the negative impact of *facial makeup* on the matching performance of four face recognition algorithms. Their experiments suggested a significant decrease in matching accuracy when comparing facial images before and after the application of cosmetics. The use of makeup as a face alteration method poses a significant challenge to biometric systems, since it represents a simple, non-permanent, and cost effective way of confounding the system. Further, the use of makeup is socially acceptable in many parts of the world. Thus, detecting the presence of makeup in a face image can benefit face recognition systems from the perspective of both security (by flagging face spoofing or obfuscation attempts[1]) and recognition accuracy (by facilitating the application of makeup-specific preprocessing routines). Additionally, automated age estimation and aesthetic prediction methods can utilize knowledge about the presence of makeup to refine their outputs.

In this work, we design a method to detect facial makeup in unconstrained face images. Given a face image, the proposed method first extracts a set of features based on shape, color and texture. This feature set is then used by a classifier to detect the presence or absence of makeup in the input face image. Experiments are conducted on two challenging and unconstrained datasets containing images of female subjects. The datasets include variations in facial pose, illumination, expression, and image resolution. Further, we use the output of the makeup detector to selectively pre-process face images prior to matching makeup images against no-makeup images. The proposed approach is observed to improve the matching performance of face recognition.

The rest of the paper is organized as follows. Sections 2 and 3 describe the visual impact of makeup on the face and introduce the proposed makeup detection method. Section 4 introduces the databases that were assembled for this study. Section 5 presents experiments validating the effectiveness of the proposed method in detecting makeup. Section 6 introduces a face recognition scheme that exploits knowledge of the presence of makeup to selectively pre-process face images prior to matching. Finally, Section 7 concludes the paper and discusses future directions.

---

[1]*Spoofing* entails the use of makeup to look like another person. *Obfuscation* entails the use of makeup to mask one's own identity

Table 1. Examples of face altering makeup items.

Face region	Related makeup item
Eye region	kohl, mascara, eye shadow, false eyelashes, eyebrow pencils, creams, waxes, gels and powders
Lip region	lipstick, lip gloss, liner, plumper, balm
Global skin appearance	concealer, foundation, face powder, rouge, blush or blusher, contour powder/creams, highlight, bronzer

2. Makeup detection

Facial makeup is commonly used to enhance the aesthetics of a face, although it can also be used for concealing scars, moles and tattoos. In a recent British poll of 2,000 women[2], more than half the subjects reported wearing makeup every day, with almost two thirds not leaving the house without makeup. A market research report[3] indicates that the sales volume for makeup in the United States was 3.6 Billion in 2011 - a 9% increase from 2010. This suggests that the use of makeup is widespread and has become a daily necessity for many. While makeup consumers are predominantly female, the beauty market has been increasingly producing products geared toward a male clientele.

Different types of makeup can be applied to different regions of the face. Table 1 gives a few examples. Makeup can fall under two categories:

- Light makeup (see Fig. 1(a)): The makeup cannot be easily perceived, since the applied colors correspond to natural skin, lip and eye colors.

- Heavy makeup (see Fig. 1(b)): The makeup is clearly perceptible (e.g. red or dark lips, strongly accentuated eyes).

The notion of light or heavy makeup does not necessarily relate to the number of makeup products that were used, but rather to the difference in facial appearance before and after applying makeup (Fig. 1).

The aesthetic effects induced by makeup are a consequence of perceptual changes in facial appearance, which can be attributed to altered facial feature shapes due to contouring, contrast changes in the mouth and eye region, and refined skin texture and color, as can be seen in Fig. 1. From a 2D image processing perspective we note that makeup can change the shape, texture and color information of global and local facial features. Therefore, it is essential to utilize both global and local information when detecting the presence of makeup.

---

[2]http://www.superdrug.com/content/ebiz/ superdrug/stry/cgq1300799243/survey release - jp. pdf

[3]https://www.npd.com/wps/portal/npd/us/news/ press-releases/pr_120301/

(a) Subject A   (b) Subject B

Figure 1. The before and after makeup images of two subjects. The after makeup images exhibit color, shape and texture changes in the eye and mouth regions.

2.1. Colorspace

One of the key aspects of the proposed approach is the choice of colorspace used to process the images. Based on visual assessment of makeup images in various colorspaces such as RGB (Red/Green/Blue), Lab (Luminance/Chromatic Components), and HSV (Hue/Saturation/Value), we decided to use the HSV colorspace. As can be seen in Fig. 2, information pertaining to makeup can be better discerned in the saturation channel of the HSV colorspace. The HSV color space is a non-linear transform of the RGB space and is given by ([15]):
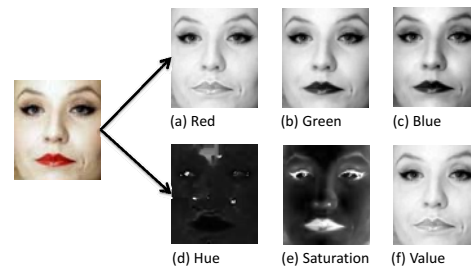
$$H = \arctan\left\{\frac{\sqrt{3}(G-B)}{(R-G)+(R-B)}\right\}, S = 1 - \frac{\min\{R,G,B\}}{V},$$
$$V = \frac{(R+G+B)}{3}.$$

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

(d) Hue   (e) Saturation   (f) Value

Figure 2. Visualizing a face as individual channels in the RGB (top) and HSV (bottom) colorspaces.

2.2. Proposed method

To the best of our knowledge, the only work related to automatic makeup detection is a very recent study by Varshovi [13], which was tested on 120 images of 21 *frontal, neutral* expression female subjects and obtained classification accuracies of 90.62% for eye-shadow detection, 93.33% for lip-stick detection and 52.5% for liquid foundation detection. The study explored texture and color features for makeup cues.

We pose the problem of makeup detection as a two-class pattern classification problem. The makeup detector has the following components: (a) Face detection and landmark localization; (b) Face normalization; (c) ROI Extraction; (d)

Feature Extraction; (e) Feature Classification (see Fig. 3). The AdaBoost face detector in OpenCV is used to provide an approximate location and scale of the face in the input image. Feature landmarks are then estimated within the facial region based on the method in [6] which employs a generative model for the landmark points and a discriminative model for the landmark appearance. The generative model is a Gaussian Mixture Model (GMM) that characterizes the joint probability distribution of landmark positions. The discriminative model consists of Haar-like filters and an AdaBoost classifier for locating and characterizing the appearance of each landmark. This is followed by face cropping and alignment based on the detected eye landmarks (estimated from the positions of left and right corner eye landmarks). For further processing, we consider this cropped facial area, as well as three specific regions of interest (ROIs): the regions around the left eye, the right eye and the mouth. Next, a set of shape, color and texture features are extracted from the face and ROIs (only color features are extracted from ROIs at this time), and a trained classifier is used to classify the extracted features into one of two classes: makeup or no-makeup. The proposed framework is illustrated in Fig. 3 and explained in detail below.
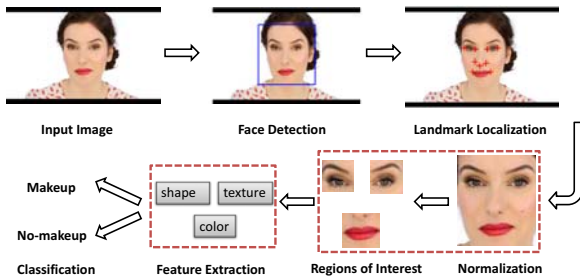

Figure 3. Proposed framework for automatic facial makeup detection.

2.3. ROI detection

After face detection and landmark localization, we geometrically normalize the face images using an affine transformation in order to remove variations due to scale and pose. All normalized face images are cropped and resized to a dimension of 150×130 pixels. Then the three ROIs are localized at pre-defined locations in the resized image and have the following dimensions: Left eye ROI: 52×52; Right eye ROI: 52×52; Mouth ROI: 56×62. Examples of these ROIs can be seen in Fig. 4.

2.4. Feature extraction

The proposed features for makeup detection are based on shape, texture and color descriptors. The choice of features was based on the following observations: (a) Visually, the dominant impact of makeup is on the color attributes of a

(a) Right eye ROIs

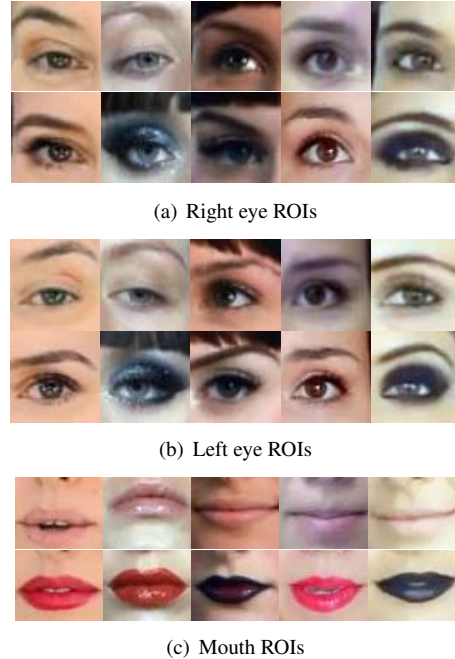(b) Left eye ROIs

(c) Mouth ROIs

Figure 4. Examples of eye and mouth ROIs of 5 subjects. Top row: Without makeup. Bottom row: With makeup.

facial image. Therefore, color-based features are used. (b) Since local shape and texture information are impacted by makeup, a set of Gabor filters are used to extract shape and texture information across different spatial scales and filter orientations. (c) Makeup can alter small-scale features in faces. Therefore, the Local Binary Pattern (LBP) operator is used to capture micro-pattern details of the facial image.

Below, we give details about the extracted features that are based on by Zhu et al. [16].

2.4.1 Color descriptor

Color is a prominent low-level visual feature that can be used to describe images [15]. To extract color-based features, we first tessellate each ROI into 5×5 non-overlapping blocks and then compute color moments within every block (Fig. 5(a) and 5(c)). Let $I_{x,y}$ denote an image pixel at (x, y) within a block in one of the channels. If N is the total number of pixels, then the first order moment (mean) is calculated as $\rho = \sum_{x,y} \frac{1}{N} I_{x,y}$; the second order moment (standard deviation) as $\sigma = \sqrt{\frac{1}{N} \sum_{x,y} (I_{x,y} - \rho)^2}$; and the third order moment (skewness) as $\gamma = \sqrt[3]{\frac{1}{N} \sum_{x,y} (I_{x,y} - \rho)^3}$. These features are extracted from all 3 channels resulting in a 225-dimensional feature vector. To extract color moments from the entire face image, the face is partitioned into 9 non-overlapping block regions, resulting in an 81-dimensional color feature vector, as illustrated in Fig. 5(b) and 5(d).
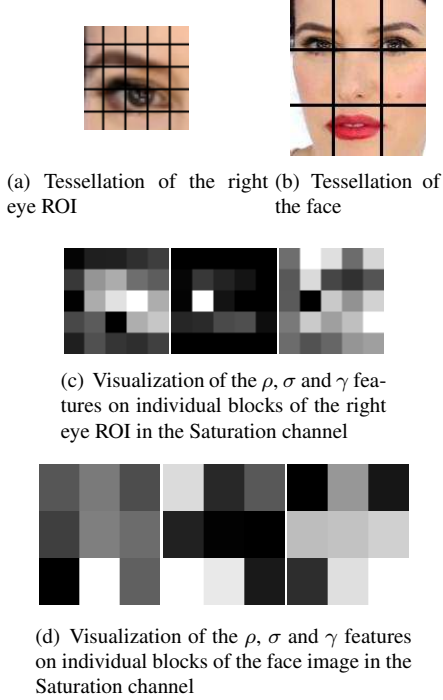
(a) Tessellation of the right eye ROI (b) Tessellation of the face

(c) Visualization of the ρ, σ and γ features on individual blocks of the right eye ROI in the Saturation channel

(d) Visualization of the ρ, σ and γ features on individual blocks of the face image in the Saturation channel

Figure 5. The tessellation scheme used for extracting color-based features.

2.4.2 Shape descriptor

Three types of shape descriptors were used to extract additional features from the entire face. These are described below.

The first descriptor is based on *Gabor wavelets* and the same parameter settings specified in [9] were adopted. The size of each Gabor kernel was 64×64. Upon convolving the input face with the set of Gabor filters, we obtain 40 image outputs. We then calculate the mean, variance, and skewness for each of these images resulting in 120 features.

We utilize a second shape descriptor known as *GIST*, which was originally designed by Torralba and Oliva [11] for scene structure representation. It first applies pre-filtering to reduce illumination variations thereby preventing some local image regions from dominating the energy spectrum. The pre-filtering can be denoted as follows:

$$I'(x,y) = \frac{I(x,y) \times h(x,y)}{\epsilon + \sqrt{[I(x,y) \times h(x,y)]^2 \times g(x,y)}}, \quad (1)$$

where $I(x,y)$ is the input, $g(x,y)$ is a low-pass Gaussian filter and $h(x,y) = 1 - g(x,y)$ is the corresponding high-pass filter. Discrete Fourier Transform (DFT) is then applied to a set of Gabor filtered images (4 scales and 8 orientations) and the resultant image is divided into blocks by a 4×4 grid, from which the mean moment is extracted from each block. This results in a GIST feature vector of length

$4 \times 4 \times 32 = 512$.

The third shape descriptor is based on *edge information*. Since the application of eye and mouth makeup enhances the local edge structure (corner and contour), an edge orientation histogram is computed. A Canny edge detector is first applied to obtain the edge map, from which an edge orientation histogram (EOH) [16] is extracted based on a 37-bin quantization of edge orientation values (see Fig. 6).

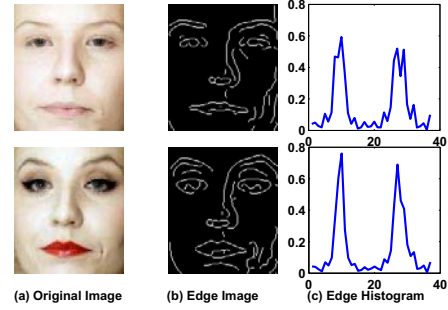(a) Original Image (b) Edge Image (c) Edge Histogram

Figure 6. Examples of edge orientation histograms for the same subject with and without makeup. The application of makeup typically increases edge information around the eyes and the mouth.

2.4.3 Texture descriptor

The LBP texture descriptor [1] is used to characterize micro-patterns or micro-structures in the face image by binarizing local neighborhoods based on the differences in pixel intensity between the center pixel and neighborhood pixels, and converting the resulting binary string into a decimal value. Uniform LBP patterns (refers to those binary patterns that have at most two bitwise transitions from 0 to 1 or 1 to 0) are extracted, resulting in a 59-bin histogram feature vector (58 out of 256 patterns are uniform when a neighborhood size of 8 is used). Uniformity is an important aspect as it characterizes micro-features such as lines, edges and corners, which are enhanced by the application of makeup.

The overall dimension of the feature vector, which integrates the color, shape and texture features from both face and ROIs, is 1484. Separately, the dimensionalities of the face feature vector and the ROI feature vectors are 809 and 675, respectively (see Table 2). Each feature dimension is normalized to zero mean and unit variance.

3. Classification

A pattern classifier, trained on labeled data, is used to classify the feature vector into one of two classes: "makeup" or "no-makeup". We utilized the SVM [4] and

Table 2. The dimensionality of features used in makeup detection.

Attribute	Feature	Face-Dim	ROI-Dim
Color	Moments	81	225×3
Shape	Gabor	120	
	GIST	512	-
	EOH	37	
Texture	LBP	59	-
Total		**809**	**675**

Adaboost [2] classifiers in this work [4].

SVM: Support Vector Machine (SVM) searches for a linear boundary that maximizes the margin between two classes of patterns by solving the following optimization problem:

$$\min_{w,\varepsilon} \left\{ \frac{1}{2} ||w||^2 + C \sum_{i=1}^{N} \varepsilon_i \right\}, \quad (2)$$

subject to the constraint: $y_i(w^T \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0$. Here, b is the bias and w is the weight vector, $(\mathbf{x_i}, y_i)$ is the labeled i^{th} training sample, ε_i is a variable introduced to control the trade off between a large margin and a small error penalty, C is a constant and N is the total number of training samples. The Gaussian RBF kernel is used (to compute ϕ) and defined as: $k(\mathbf{x}_i, \mathbf{x}_j) = exp(-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2), \gamma > 0$. The kernel is related to the transform ϕ by the equation $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \cdot \phi(\mathbf{x}_j)$. The optimum values for C and the kernel parameter γ are obtained by a grid-search of the parameter space based on the training set.

Adaboost: The principle of Adaboost is to combine multiple weak classifiers to form a single strong classifier as $y(\mathbf{x}) = \sum_{t=1}^{T} \alpha_t h_t(\mathbf{x})$, where $h_t(\mathbf{x})$ refers to the weak classifiers operating on the input feature vector $\mathbf{x}$, T is the number of weak classifiers, α_t is the corresponding weight for each weak classifier and $y(\mathbf{x})$ is the classification output. In this work, for every pair of feature values (f_i, f_j) in the feature vector $\mathbf{x}$, five types of weak binary classifiers are defined:

$$h_t(\mathbf{x}) \equiv g_k(f_i, f_j) = 1, \quad \text{if } d(f_i - f_j) > t_k, \quad (3)$$

where $k = 1 \ldots 5$, $t_1 = 0$, $t_2 = 5$, $t_3 = 10$, $t_4 = 25$ and $t_5 = 50$. By changing the inequality sign from $>$ to $<$, another five types of weak classifiers are generated, resulting in a total of 10 types of weak classifiers. Since g_k is non-commutative, $g_k(f_i, f_j) \neq g_k(f_j, f_i)$, the total number of weak classifiers for a pair of features f_i and f_j is 20.

Each pair-wise comparison results in a binary value, which is used as a weak-classifier by the Adaboost algorithm. For the 1484 dimensional feature vector, Adaboost

will generate a set of $T = \binom{1484}{2} \times 20 = 22,007,720$ weak classifiers. For the 809 and 675 dimensional feature vectors extracted from face and ROIs, respectively, the number of weak classifiers are $T = \binom{809}{2} \times 20 = 6,536,720$ and $T = \binom{675}{2} \times 20 = 4,549,500$. After performing feature selection and weighting (for estimating α_t) based on the classical Viola-Jones scheme, a total of 1000 weak classifiers are retained in each case.

4. Makeup databases

4.1. YouTube makeup database (YMU)

In this study, we utilized the database introduced by Dantcheva et al. [5] which contains the before and after makeup images of 151 Caucasian female subjects taken from YouTube makeup tutorials (99 subjects were used in their work in [5]). Examples are shown in Fig. 7 (after face cropping and alignment). For a majority of the subjects there are four shots per subject - two shots before the application of makeup and two shots after the application of makeup. For some subjects, there is either only one shot or three shots each before and after the application of makeup. The total number of images in the dataset is 600, with 300 makeup images and 300 no-makeup images. We note that the degree of makeup in this database varies from subtle to heavy. The database is relatively unconstrained, exhibiting variations in facial expression, pose and resolution.

Figure 7. Facial images showing variations in pose, illumination, expression and resolution from the YMU database [5]. The corresponding eye and mouth ROIs are shown in Fig. 4.

4.2. Makeup in the wild database (MIW)

In addition to the aforementioned dataset, we assembled another database of 154 images (77 with makeup, and 77 without makeup) corresponding to 125 subjects. Since the images are obtained from the Internet, we refer to this database as Makeup in the "Wild". A few examples are shown in Fig. 8[5]. The purpose of using this database is to evaluate the generalization capability of the proposed makeup face detector where the training is performed using the YMU database and testing is done on the MIW database.

---

[4]A number of other classifiers were also experimented with. SVM and AdaBoost resulted in the best performance and are reported here.

[5]Images from the MIW database are available in the authors' webpage: http://www.antitza.com/makeup-datasets.html

In both the databases, an image is labeled as "makeup" even if cosmetic details are present in only a portion of the face.

Figure 8. Sample images from the MIW database. Images are collected from the Internet. Top row shows images without makeup and the bottom row shows images with makeup. Note the unconstrained nature of the images.

5. Experiments

In order to evaluate the performance of the proposed makeup detector, we employ a 5-fold cross-validation scheme. Here, the YMU dataset is divided into 5 folds with approximately 30 subjects in each fold. 4 folds are used for training the makeup detector, and the remaining fold is used for testing it. This is repeated 5 times. **Note that the subjects in the training set are not present in the test set.** The performance of the makeup detector is reported using two metrics: (a) Classification Rate (CR): The percentage of makeup and no-makeup images that are correctly classified by the detector; (b) Receiver Operating Characteristic (ROC) curve: Here, the true positive rate (TPR: the percentage of "makeup" images that are correctly classified as "makeup") is plotted as a function of the false positive rate (FPR: the percentage of "no-makeup" images that are incorrectly classified as "makeup").

5.1. Makeup detection

In this section, we evaluate the performance of the proposed makeup detection system on the YouTube database. First, we investigate the performances of individual ROIs for makeup detection. As presented in Table 3, when extracting features from the entire face region, a classification rate of 87.25% using SVM was obtained. The left and right eye ROIs achieve classification rates of 81.71% and 80.68%, respectively. The mouth ROI has the lowest classification rate of 58.94%. This could be due to the inability of the color moments to capture the homogeneous region created by the application of lipstick. The SVM classification results for individual face feature sets are as follows: color moments: 77.62%; Gabor: 62.08%; GIST: 86.79%;

Table 3. The classification rates of the SVM-based and Adaboost-based makeup detector on the YMU database.

ROI	SVM (%)	Adaboost (%)
Entire Face	87.25 ± 1.91	88.98 ± 3.54
Left eye	81.71 ± 4.67	75.72 ± 1.99
Right eye	80.68 ± 3.08	79.89 ± 4.22
Mouth	58.94 ± 3.47	57.46 ± 5.94
Left eye + Right eye + Mouth	87.62 ± 2.01	85.83 ± 3.84
Face + Left Eye + Right Eye + Mouth	$\mathbf{91.20 \pm 0.56}$	$\mathbf{89.94 \pm 1.60}$

EOH: 56.68%; LBP: 50.78%. When fusing the three 225-dimensional feature vectors corresponding to the individual ROIs, the classification rate increases significantly to 87.62%. When the entire 1484-dimensional feature vector is used, the performance further increases to 91.2%. The ROC curves corresponding to this experiment are reported in Fig. 9. We note that the area under the curve (AUC) is rather high in all five trials, indicating the efficacy of the proposed makeup detector. The classification rate for each trial in the 5-fold cross-validation experiment is reported in Table 4. Here, the Adaboost classifier obtains an average accuracy of $89.94 \pm 1.60\%$, which is slightly lower than the SVM-based classifier ($91.20 \pm 0.56\%$).

Table 4. Classification Rates of the SVM and Adaboost classifiers on the YMU database. The numbers in parentheses indicate the number of "no-makeup" and "makeup" images in each trial.

Trial	Train	Test	SVM (%)	Adaboost (%)
1	487 (243/244)	113 (57/56)	92.04	91.15
2	473 (237/236)	127 (63/64)	90.55	87.40
3	487 (244/243)	113 (56/57)	91.15	90.27
4	457 (228/229)	143 (72/71)	90.91	89.51
5	496 (248/248)	104 (52/52)	91.35	91.35
Average			**91.20**	**89.94**

Next, the proposed makeup detector that is trained on the YMU database (all 600 images), is tested on the MIW database. Sample outputs are presented in Fig. 10. The face detector failed in 22 of the 154 images. A classification rate of 95.45% for SVM and 92.21% for Adaboost was obtained. At 1% FPR, a TPR of 93.51% and 84.42% was obtained for SVM and Adaboost, respectively. The corresponding ROC curves are shown in Fig. 9(f). This confirms the generalization ability of the proposed approach.

Experiments were conducted using Matlab R2009a on a 32 bit windows operating system with Intel Core i7-2600s CPU at 2.80GHz and 3.16GB RAM. The makeup detector processes a face image in 0.78 seconds.

(a) Trial 1: $C = 128.0, \gamma = 1.2 \times 10^{-4}$ (b) Trial 2: $C = 128.0, \gamma = 1.2 \times 10^{-4}$ (c) Trial 3: $C = 8.0, \gamma = 0.002$

(d) Trial 4: $C = 32.0, \gamma = 0.002$ (e) Trial 5: $C = 32.0, \gamma = 4.8 \times 10^{-4}$ (f) MIW dataset: $C = 8.0, \gamma = 0.002$
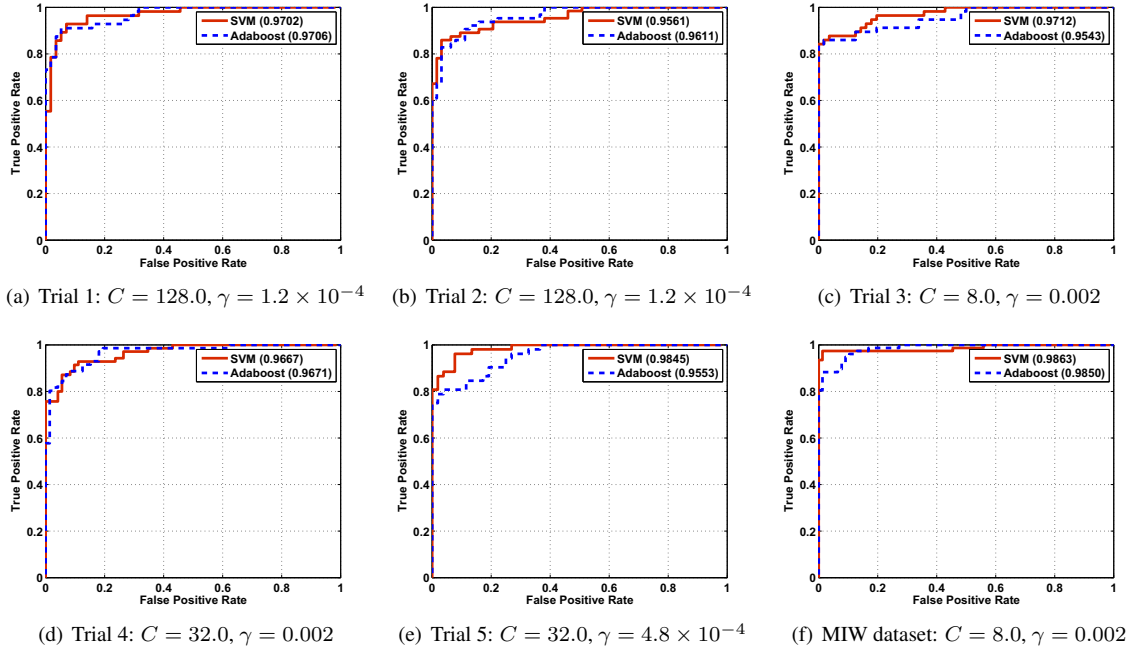
Figure 9. ROC curves of SVM-based and Adaboost-based makeup detectors on the YMU database for all 5 trials in the cross-validation scheme [(a) - (e)] and on the MIW database [(f)]. The parameters used by the SVM is indicated below each graph. The numbers within parentheses in each legend indicate the AUC values.
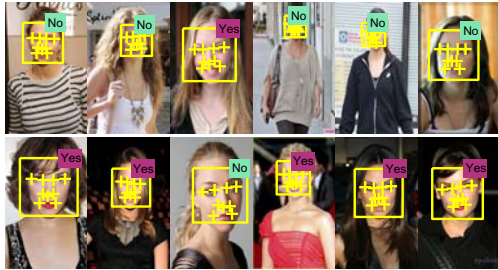

Figure 10. Output of the proposed makeup detector on the MIW images shown in Fig. 8. The detector was trained using all the images from the YMU database [5].

6. Application in face recognition

In this section, we discuss the use of the proposed SVM-based makeup detector in the context of face recognition. In [5], the authors showed that the recognition performance of face matchers decreases when matching makeup images ($\mathcal{M}$) against their no-makeup counterparts ($\mathcal{N}$). In order to address this issue, we devise a pre-processing routine. The idea here is to suppress the effect of makeup by utilizing a photometric normalization routine along with a blurring operator that smoothens the edge-like features induced by makeup. Specifically, if one of the two images to be matched is deemed to have makeup and the other is deemed to have no makeup, then both images are photometrically normalized using the Multiscale Self Quotient Image (MSQI) technique[6] before they are input to the matcher. The self-quotient image, Q, of image I is defined as [14]: $Q = \frac{I(x,y)}{\hat{I}(x,y)} = \frac{\rho_w(x,y)n(x,y)s}{G_k * [\rho_w(x,y)n(x,y)s]}$, where $\rho_w(x,y)$ is the albedo of the facial surface, n is the surface normal, s is the lighting reflection, G_k is the weighted Gaussian smoothing filter and k is the size of the kernel. In this work, four different kernel sizes were used (multi-scale): $3 \times 3, 5 \times 5, 11 \times 11, 15 \times 15$. The output image is the summation of the 4 filtered images. The corresponding sigma values used by the Gaussian filter were 1, 1.2, 1.4 and 1.6, respectively.

To test the efficacy of the scheme, we again use the YMU database and adopt the same 5-fold cross-validation scheme for evaluating performance. The Multi-Scale LBP (MSLBP) method is used for encoding and matching face images [12]. The MSLBP operator is very similar to the LBP operator (see Section 2.4.3) but with two main differences: (a) the binary pattern of a pixel is computed by comparing the mean values of sub-blocks; (b) the binary pattern is computed at multiple scales and over a dense grid with spacing of 10 pixels. For a specific scale s, the size of the sub-block considered is $\frac{s}{3} \times \frac{s}{3}$. We consider 4 different scales in this work: 3, 9, 15, 21. For each scale a uniform

---

[6]http://luks.fe.uni-lj.si/sl/osebje/vitomir/
face_tools/INFace/index.html

LBP histogram is generated and the concatenated histogram values across the 4 scales serves as a feature vector for the face. Two such feature vectors are compared using the Histogram Intersection Distance to generate a match score. For each of the 5 trials we report results on the 3 matching scenarios suggested in [5]: no-makeup vs no-makeup images ($\mathcal{N}$ vs $\mathcal{N}$); makeup vs makeup images ($\mathcal{M}$ vs $\mathcal{M}$); makeup vs no-makeup images ($\mathcal{M}$ vs $\mathcal{N}$). In Table 5, the face verification rate is reported at a False Accept Rate (FAR) of 1%. "Aggregate" refers to the computation of verification results after pooling the match scores from all 5 trials for each scenario. *It has to be noted that the application of the pre-processing routine increases the verification performance for the $\mathcal{M}$ vs $\mathcal{N}$ case, without impacting the accuracy of the $\mathcal{M}$ vs $\mathcal{M}$ and $\mathcal{N}$ vs $\mathcal{N}$ cases.*

If pre-processing is applied to **all** image pairs before matching, then the corresponding verification results for $\mathcal{M}$ vs $\mathcal{N}$, $\mathcal{M}$ vs $\mathcal{M}$ and $\mathcal{N}$ vs $\mathcal{N}$ are 54.78%, 85.43% and 92.72%, respectively. While these results are comparable to the ones reported in Table 5, note that selective pre-processing as proposed in this work would (a) detect the presence of makeup and (b) avoid applying the pre-processing routine to all image pairs. The former is potentially useful information in identifying spoofing/obfuscation attempts. It must be noted that these are preliminary results, but do convey the applicability of the proposed makeup detector in face recognition.

Table 5. Face verification performance (%) at a FAR of 1% before and after (B/A) applying the proposed face pre-processing scheme. The pre-processing scheme is invoked only when one image is deemed to have makeup and the other image is deemed to be without makeup by the proposed makeup detector. Column 3 highlights the improvement in verification results for the $\mathcal{M}$ vs $\mathcal{N}$ case.

Trial	$\mathcal{M}$ vs $\mathcal{N}$	Increase	$\mathcal{M}$ vs $\mathcal{M}$	$\mathcal{N}$ vs $\mathcal{N}$
1	56.25/65.55	9.30	92.86/92.86	96.43/96.43
2	52.75/55.64	2.89	73.44/80.47	87.69/87.76
3	48.54/54.00	5.46	83.33/83.33	89.29/89.29
4	45.55/49.23	3.68	80.00/80.00	92.97/95.74
5	54.34/56.35	2.01	88.46/88.46	95.73/96.15
Aggregate	**48.88/54.10**	**5.22**	**84.70/86.05**	**92.72/92.72**

7. Summary and future work

In this paper, we proposed an automated makeup detector for unconstrained facial images. The proposed detector utilizes shape, texture and color features extracted from the entire face, as well as facial subregions, to determine the presence of makeup. Experiments conducted on two unconstrained face datasets resulted in makeup detection rates of up to 93.5% (at 1% false positive rate) and overall classification rates of up to 95.45%. The output of the makeup detector was then used to perform adaptive pre-processing in the context of face recognition. Experimental results indicate that applying the proposed pre-processing routine can improve the recognition accuracy of face matchers when matching makeup images against no-makeup images. However, more work is necessary in this regard. Future work will involve improving the performance of the makeup detector and exploring methods to remove artifacts introduced by the application of makeup. Specifically, we are interested in the problem of determining the *degree* of makeup applied to the face - this will have benefits in obfuscation/spoofing scenarios. Further, we will test the proposed method on datasets that include male subjects. Finally, the makeup detector can be used to refine the output of automatic age estimation and beauty assessment algorithms that may also be impacted by the application of makeup.

References

[1] T. Ahonen, A. Hadid, and M. Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Trans. on PAMI*, 28(12):2037–2041, 2006.

[2] J. Bekios-Calfa, J. M. Buenaposada, and L. Baumela. Revisiting linear discriminant techniques in gender recognition. *IEEE Trans. on PAMI*, 33(4):858–864, 2011.

[3] H. S. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa. Recognizing surgically altered face images using multiobjective evolutionary algorithm. *IEEE Trans. on Information Forensics and Security*, 8(1):89–100, 2013.

[4] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. on Intelligent Systems and Technology*, 2(3):1–27, 2011.

[5] A. Dantcheva, C. Chen, and A. Ross. Can facial cosmetics affect the matching accuracy of face recognition systems? In *BTAS*, 2012.

[6] M. Everingham, J. Sivic, and A. Zisserman. "Hello! my name is... buffy" – automatic naming of characters in TV video. In *BMVC*, pages 899–908, 2006.

[7] Y. Fu, G. Guo, and T. S. Huang. Age synthesis and estimation via faces: a survey. *IEEE Trans. on PAMI*, 32(11):1955–1976, 2010.

[8] S. Z. Li and A. K. Jain, editors. *Handbook of Face Recognition, 2nd Edition*. Springer, 2011.

[9] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Trans. on Image Processing*, 11(4):467–476, 2002.

[10] J. Maatta, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using texture and local shape analysis. *IET Biometrics*, 1(1):3 –10, 2012.

[11] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[12] S. Paris, H. Glotin, and Z. Zhao. Real-time face detection using integral histogram of multi-scale local binary patterns. In *ICIC*, pages 276–281, 2011.

[13] S. Varshovi. Facial makeup detection using HSV color space and texture analysis. Master's thesis, Concordia University, Canada, 2012.

[14] H. Wang, S. Z. Li, Y. Wang, and J. Zhang. Self quotient image for face recognition. In *ICIP*, pages 1397–1400, 2004.

[15] H. Yu, M. Li, H. Zhang, and J. Feng. Color texture moments for content-based image retrieval. In *ICIP*, pages 929–932, 2002.

[16] J. Zhu, S. C. Hoi, M. R. Lyu, and S. Yan. Near-duplicate keyframe retrieval by nonrigid image matching. In *ACM Multimedia*, pages 41–50, 2008.