Real-time Embedded Age and Gender Classification in Unconstrained Video

Ramin Azarmehr, Robert Laganière, Won-Sook Lee
School of Electrical Engineering and Computer Science
University of Ottawa
Ottawa, ON K1N 6N5 Canada
{razar033,laganier,wslee}@uottawa.ca

Christina Xu, Daniel Laroche
CogniVue Corporation
Gatineau, QC, Canada
{cxu,dlaroche}@cognivue.com

Abstract

In this paper, we present a complete framework for video-based age and gender classification which performs accurately on embedded systems in real-time and under unconstrained conditions. We propose a segmental dimensionality reduction technique using Enhanced Discriminant Analysis (EDA) to reduce the memory requirements up to 99.5%. A non-linear Support Vector Machine (SVM) along with a discriminative demographics classification strategy is exploited to improve both accuracy and performance. Also, we introduce novel improvements for face alignment and illumination normalization in unconstrained environments. Our cross-database evaluations demonstrate competitive recognition rates compared to the resource-demanding state-of-the-art approaches.

1. Introduction

Recently, automatic demographic classification has found its way into industrial applications such as surveillance monitoring, security control, and targeted marketing systems. Implementing a demographic classifier on embedded platforms can extend its usefulness to even a wider variety of applications in mobile services. Ng *et al*. [1] surveyed potential embedded applications such as human-robot interaction, or gender recognition to speed-up face recognition on mobile devices. Electronic Customer Relationship Management (ECRM) [2] is another fast-growing technology that facilitates marketing customized products and services based on customer's age or gender in an automatic and non-intrusive way. Many of such systems demand a robust and real-time demographic classifier that is able to process 15 to 25 frames per second (fps). To achieve this, the arising challenge is the constrained memory and computation power of the embedded systems.

Training with 200,000 images, Irick *et al*. [3] implemented an *appearance-based* gender classifier on FPGA using neural networks, achieving 83% accuracy on a database of 3,826 images. Utilizing a Support Vector Machine (SVM) with Radial Basis Function (RBF), Moghaddam *et al*. [4] reported 96.6% recognition rate for classifying gender on 1,775 images of FERET database [5]. However, with a cross-database evaluation, Baluja *et al*. [6] achieved only 93.5% accuracy using the same approach. Beikos-Calfa *et al*. [7] proposed a holistic but resource-intensive strategy that employed Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) for gender recognition, and reported 93.33% accuracy. Later, Fazl-Ersi *et al*. [8] proposed a feature-based method using SVM and Local Binary Pattern (LBP) operator [9], and achieved the recognition rate of 91.59% for gender, and 63.01% for age classification on Gallagher [10] dataset.

Typically, in holistic approaches, a single large feature vector is meant to feed the classifier, but the bulky nature of this vector is at odds with the limited resources of embedded platforms. Moreover, the high degree of redundancy and presence of textural noise can degrade the accuracy of classification. In this paper, we present practical solutions to these problems in order to enable the implementation of a real-time age and gender classifier on the resource-limited embedded platforms.

Our contributions are summarized into different sections as follows: We start by proposing an improvement in face alignment using the nose in Section 2.1, and a robust illumination normalization strategy in Section 2.2. A review of local patterns and our further optimizations are presented in Section 2.3. Next, a segmental dimensionality reduction method for multi-resolution feature vectors is introduced in Section 2.4. We generalize a discriminative demographics classification approach in Section 2.5 to improve the performance. Finally, we present our experimental setup, results and conclusions in Sections 3, 4, and 5, respectively.

2. Methodology

Generally, the face classification methods are sensitive to face localization errors and variations in illumination. Therefore, the face image should be normalized prior to fea-

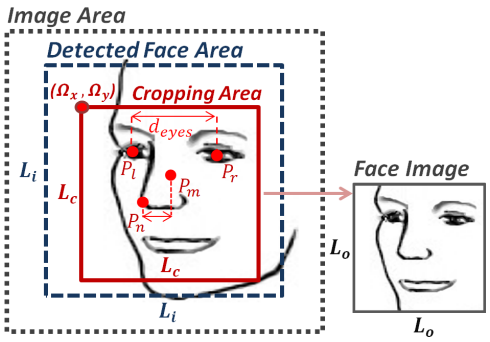
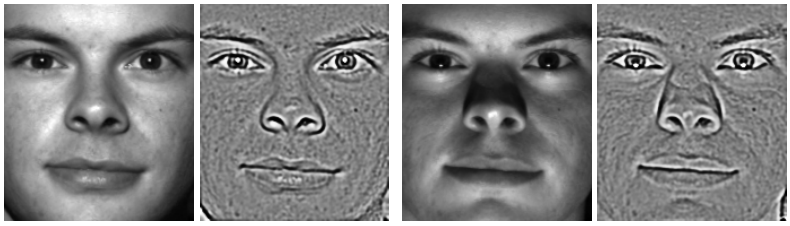


Figure 1. Face alignment using nose and eyes.



(a) PS on Masculine Face $(5^\circ, 10^\circ)$ (b) PS on Feminine Face $(0^\circ, -35^\circ)$

Figure 2. The effect of illumination on gender perception of a male subject. Original images [11] illuminated from (azimuth, elevation). Masculine look after applying Pre-processing Sequence (PS) [12] on both faces.

ture extraction. On the other hand, in our application the limitations of embedded systems in terms of memory and computational requirements must be taken into consideration. In here, we investigate these issues and present robust solutions for them.

2.1. Face Alignment

A popular approach in face alignment is the positioning of frontal face images into an upright canonical pose based on the position of eyes [13]. To locate the eyes, we use the open-source *flandmark* library [14] which is a memory efficient, real-time, and fairly accurate facial landmark detector. Figure 1 illustrates some detected facial landmark points on the eyes and nose. The eyes can be aligned horizontally by an in-plane rotation of the face image into an upright pose using the angle $\theta = \arctan\left(\frac{P_{r,y}-P_{l,y}}{P_{r,x}-P_{l,x}}\right)$ where the points $(P_{l,x}, P_{l,y})$ and $(P_{r,x}, P_{r,y})$ denote the center positions of the left and right eye.

Typically, the distance between the eyes d_{eyes} is used to compute the dimensions of the cropping area where $d_{eyes} = \sqrt{(P_{r,x} - P_{l,x})^2 + (P_{r,y} - P_{l,y})^2}$. However, in uncontrolled environments as the head's *yaw* angle increases, the eyes distance shortens. As a result, the dimensions of the cropping area shrink, causing an over-scaling error proportional to the yaw angle and, consequently, the loss of information from the upper and lower parts of the face. On the other hand, as shown in Figure 1, the horizontal distance between the points P_n and P_m on the nose increases when the eyes distance d_{eyes} shortens.

Therefore, we propose to use the horizontal positions of the upper nose $P_{m,x}$ and the lower nose $P_{n,x}$ to compensate for the over-scaling in face alignment. In Equation 1, we apply the ratio of these points to find the scale factor S_0. In this work, the detected face region is an $L_i \times L_i$ square, and the resulting aligned and cropped face is an $L_o \times L_o$ square image on which the left eye is fixed at the top-left offset Ω_o. From the scale factor S_0, we compute the dimensions $L_c \times L_c$ of the cropping area with $L_c = S_0 * L_o$, its horizontal offset $\Omega_x = P_{l,x} - S_0\Omega_o$, and its vertical offset $\Omega_y = P_{l,y} - S_0\Omega_o$. Indeed, the maximum size of the cropping area is limited as a sub-region of the detected face region in order to avoid under-scaling in the case of unreasonably large distance between the points P_n and P_m.

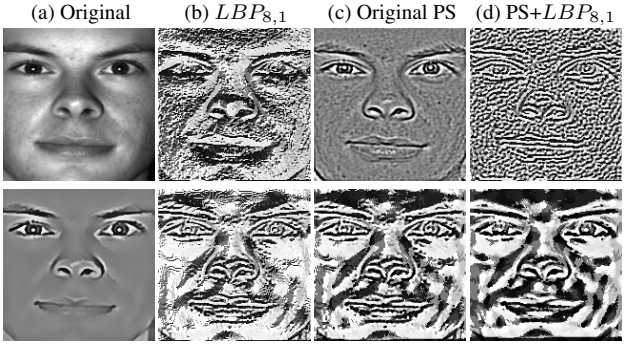
$$S_0 = \left(\frac{d_{eyes}}{L_o - 2\Omega_o}\right) * \max\left(\frac{P_{m,x}}{P_{n,x}}, \frac{P_{n,x}}{P_{m,x}}\right) \quad (1)$$

2.2. Effects of Illumination

As a matter of fact, in unconstrained environments the facial texture is prone to uneven illumination which may impact the demographics perception. Russell [15] demonstrated the Illusion of Sex on an androgynous face by only increasing the facial contrast, resulting in a feminine look on a male subject. Similarly, in our experiments we have observed the same effect on various *lighting* conditions. Figure 2 shows an androgynous male subject [11], illuminated from two different light source positions. In Figure 2(b), the light source is 35° below the horizon inducing non-monotonic gray value transformations by which the observer perceives a feminine look from the male subject. In order to normalize the photometry and reduce the effects of local shadows and highlights, we propose to apply the Pre-processing Sequence (PS) approach [12] on the aligned face image. The results of applying the PS are shown in Figures 2(a) and 2(b). Nevertheless, a large amount of textural noise is still present. We provide a practical solution to this issue in section 2.3.

2.3. Face Representation

The Local Binary Patterns (LBP) operator [9] has been widely used as a means of extracting local features of texture. Basically, for each pixel at a center of a neighborhood, the $LBP_{P,r}$ operator builds a binary sequence by applying the value of the center pixel as a threshold to P pixels in a circular neighborhood of radius r (Figure 3(b)). Typically, to reduce the redundancy and size, the *uniform* LBP operator $LBP^{u2}_{P,R}$ is used to capture the binary patterns that contain at most two bit-wise transitions from 1 to 0, or 0 to 1 [16], and the final feature vector is represented



(a) Original (b) $LBP_{8,1}$ (c) Original PS (d) PS+$LBP_{8,1}$

(e) Filtered PS (f) FPS+$LBP_{8,1}$ (g) FPS+$LBP_{8,3}$ (h) FPS+$LBP_{8,5}$

Figure 3. LBP on the original, PS, and Filtered PS (FPS) images.

by an LBP histogram (LBPH). Ahonen *et al*. [17] extended this strategy by first dividing the LBP image into J non-overlapping regions $[M_0, M_1, \ldots, M_{J-1}]$, then extracting the *local* histograms of regions, and finally concatenating the histograms into a single and spatially enhanced feature vector, as illustrated in Figure 4. Essentially, LBP operator performs robustly in the presence of monotonic intensity transformations. However, as can be seen in Figures 3(a) and 3(b), the thresholding process in LBP is highly sensitive to noise and non-monotonic transformations. A solution is to apply the Pre-processing Sequence (PS) normalization prior to LBP (Section 2.2). Surprisingly, as shown in Figures 3(c) and 3(d), the PS only intensified the negative effects of LBP noise, and tuning its default parameters could not improve the results.

To suppress the noise, we propose to add a *Bilateral filtering* stage to the PS approach. Unlike Gaussian filter, a bilateral filter can effectively suppress the noise while preserving important image features like edges. It is noteworthy that, as advised in [18], we apply the bilateral filtering in two separate iterations: before and after the PS approach. Filtering the image in Figure 3(c), we obtain the photometrically enhanced image in Figure 3(e). As a result, the corresponding LBP images are invariant to variations in illumination and noise. Figures 3(f), 3(g), and 3(h), show the LBP images extracted at three different radii from our Filtered PS image.

As a further enhancement, we employ Multi-scale Local Binary Patterns (MSLBP) [16] operator to build a scale-invariant feature vector. In our experiments, it has demonstrated its superior descriptive performance against face localization errors compared to regular LBP. The MSLBP reinforces the face descriptor by combining the histograms from multiple LBP transformations at R different radii in J regions. Equation 2 defines the uniform LBP histogram of region M_j at radius r and bin $i \in [0, L)$ [19]. Herein, L denotes the total number of bins in uniform LBP histogram. An extra bin has been added for non-uniform feature accumulation; therefore, $L = P(P-1) + 3$.

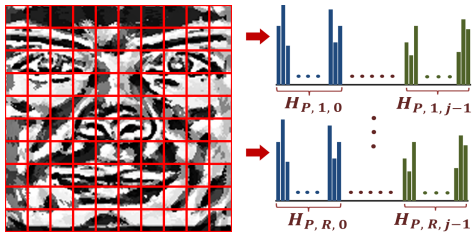


Figure 4. Extracting multi-scale local histograms

$$H^{u2}_{P,r,j}(i) = \sum_{x,y \in M_j} B(LBP^{u2}_{P,r}(x,y) = i) \qquad (2)$$

where $r \in [1, R]$, and $B(u)$ is 1 if $u \geq 0$ and 0 otherwise. Fusing R histograms at each region j, we obtain the raw face descriptor segment $\mathrm{Q}_j \in \mathbb{R}^{1\times(L.R)}$:

$$\mathrm{Q}_{j=} \left[H^{u2}_{P,1,j}, H^{u2}_{P,2,j}, \ldots, H^{u2}_{P,R,j}\right] \qquad (3)$$

$$\mathrm{Q} = \left[\mathrm{Q}_0, \mathrm{Q}_1, \ldots, \mathrm{Q}_{J-1}\right] \qquad (4)$$

In this paper, we refer to partitions of the LBP image as *regions*, and partitions of the feature vector as *segments.* The raw feature vector $\mathrm{Q} \in \mathbb{R}^{1\times(J.L.R)}$ is the ensemble of face descriptor segments for each sample, and is meant to feed the classifier's input with multi-resolution LBP features. However, its high dimensionality makes this impractical due to large time and space complexity. This so-called *curse of dimensionality* also contributes to accuracy degradation due to data redundancy and noise. Inspired by [20, 21], we minimize these problems by applying a *segmental* dimensionality reduction on each descriptor segment Q_j, separately. With respect to face recognition applications, we emphasize three major advantages in using LDA on a partitioned feature vector in demographics classification:

1. In holistic models LDA suffers from the curse of dimensionality, and a large dimensionality reduction prior to LDA can overly discard texture information. In contrast, applying LDA on separate small regions can mitigate its singularity problems while preserving important texture information.
2. In demographics classification the number of classes is finite, but theoretically, an infinite number of samples can be used to train the classifier. A low dimensional feature vector along with a large number of training samples work best to lift the curse of dimensionality from discriminant analysis.
3. Unlike face recognition, the resource-demanding Eigen-decomposition and PCA+LDA computations are only required in the training stage, and not in testing stage. We take advantage of this fact in our real-time embedded application.

2.4. Segmental Dimensionality Reduction

In general, Linear Discriminant Analysis (LDA) is a supervised reduction method that can linearly separate the classes to capture the most *discriminant* features from the face representation. It aims to maximize the ratio of between-class and within-class separability among N samples of C classes by projecting the samples into a new subspace with $C-1$ dimensions. LDA requires the dimensionality of data to be less than $N-C$ to avoid singularity problems. Herein, we have partitioned the feature vector into J smaller segments; therefore, the low dimension of the face descriptor segments Q_j can prevent singularity.

Nonetheless, the redundancy and noise in Q_j can still deteriorate the classifier's performance. In some researches [7], an oval mask is used to eliminate the background noise; however, the eyeglasses, facial expression, and the lighting and skin conditions may still influence the results. Hence, prior to LDA, we can wisely make use of PCA along with a robust feature preservation criterion in order to only retain the most *descriptive* features. PCA is formulated as a maximization problem, and its segmental projection matrix can be computed as:

$$\mathrm{W}_j^{PCA} = \underset{\mathrm{W}_j}{\operatorname{argmax}}\ \mathrm{tr}\left(\mathrm{W}_j^T \mathrm{S}_\Sigma^j \mathrm{W}_j\right) \tag{5}$$

where for each region j, $\mathrm{S}_\Sigma^j = \sum_{k=1}^{N}(\mathrm{Q}_j^k - \mu_j)(\mathrm{Q}_j^k - \mu_j)^T$ is the total scatter matrix computed from each feature segment Q_j^k of every k-th sample which are centered using the mean of all N samples $\mu_j \in \mathbb{R}^{1\times(L.R)}$.

Our criterion for eigenvector selection in PCA is that the i-th eigenvector can be preserved only if the retained energy $e_i = \frac{\sum_{m=1}^{i}\lambda_m}{\sum_{m=1}^{n}\lambda_m}$ from the first i eigenvalues λ_m is greater than a threshold τ_e [22]. This enhancement stage can be considered as an efficient *weighting* mechanism to attain more influence from more discriminative regions of face. Afterwards, the preserved information can be passed for discriminant analysis.

In LDA, we model the segmental between-class and within-class separation of samples with scatter matrices S_b^j and S_w^j, respectively. For each segment Q_j, the LDA projection matrix W_j^{LDA} can be obtained from maximizing the modified Fisher's criterion [20]:

$$\mathrm{W}_j^{LDA} = \underset{\mathrm{W}_j}{\operatorname{argmax}}\ \mathrm{tr}\left(\frac{\mathrm{W}_j^T \left(\mathrm{W}_j^{PCA}\right)^T \mathrm{S}_b^j \mathrm{W}_j^{PCA} \mathrm{W}_j}{\mathrm{W}_j^T \left(\mathrm{W}_j^{PCA}\right)^T \mathrm{S}_w^j \mathrm{W}_j^{PCA} \mathrm{W}_j}\right) \tag{6}$$

In our method, Q_j is already low-dimensional, and N is large, so the matrix S_w^j will be non-singular. As a consequence, the matrix W_j^{LDA} can be composed from the $(C-1)$ largest eigenvectors of the matrix $\left(\mathrm{S}_w^j\right)^{-1}\mathrm{S}_b^j$ in each segment j.

An often neglected issue in using LDA for face processing applications is the generalization problem. Although a minimized within-class measure is desirable for matrix S_w^j, the within-class samples may be transformed into such a narrow region that the LDA may lose its ability to generalize test data. To prevent over-fitting and improve the numerical stability, we add a regularization term to the diagonal of S_w^j using a small positive constant γ and the same-size identity matrix I, such that $\mathrm{S}_w^j = \mathrm{S}_w^j + \gamma\mathrm{I}$ [23].

Now, to acquire the most *descriptive* and *discriminant* set of features, each segment Q_j^k of k-th sample can be projected into our *Enhanced Discriminant Analysis (EDA)* subspace $\mathrm{F}_j^k \in \mathbb{R}^{1\times(C-1)}$ using the EDA transformation matrix $\mathrm{W}_j^{EDA} \in \mathbb{R}^{(LR)\times(C-1)}$. It is noteworthy that Q_j^k must be normalized to have a zero mean, as Equation 7 illustrates.

$$\mathrm{F}_j^k = \left(\mathrm{W}_j^{EDA}\right)^T \left(\mathrm{Q}_j^k - \mu_j\right) \tag{7}$$

where $\left(\mathrm{W}_j^{EDA}\right)^T = \left(\mathrm{W}_j^{LDA}\right)^T \left(\mathrm{W}_j^{PCA}\right)^T$. Finally, we concatenate the F_j^k of all samples into a single feature matrix $\mathrm{F} \in \mathbb{R}^{N\times(J.(C-1))}$ to feed the training stage (Section 2.5). However, prior to concatenation we L2-normalize the rows of matrix F in order to provide the classifier with a coherent descriptor and regularize the similarity quantification among the samples. Needless to say, each row F^k of this matrix represents the EDA projection of the feature vector Q^k extracted from the k-th training image. In testing stage, F only has a single row representing the query image.

2.5. Classification

There exist various classification and similarity measurement techniques in LDA space, such as Euclidean or cosine distance measurement between samples. However, in this work we employ the supervised and discriminative SVM classifier [24] with an RBF kernel to guarantee an accurate classification in LDA space. Typically, a soft margin SVM with a penalty cost C_p is used to compensate for misclassification due to asymmetric class sizes and over-proportional influence of larger classes. We obtain the optimal values for

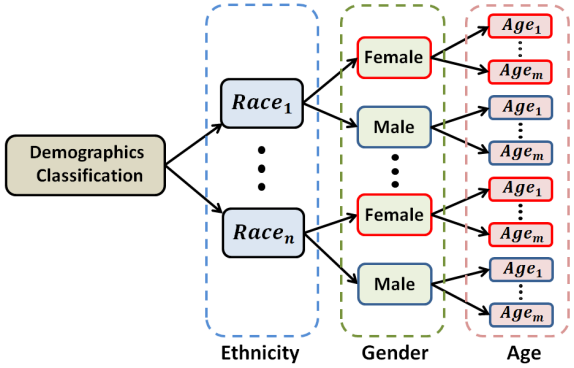


Figure 5. Demography-based discriminative classification tree

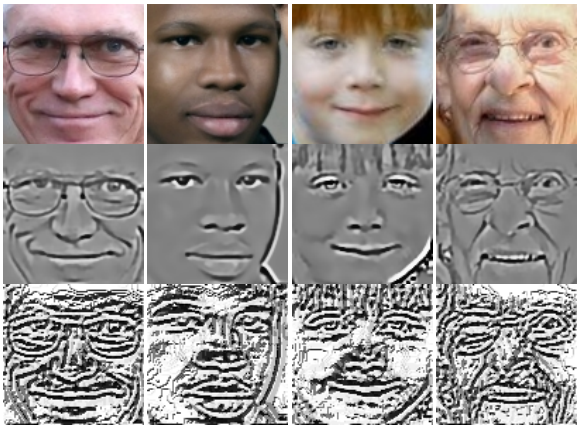


Figure 6. Our illumination normalization approach. Original images (top), Filtered PS (middle), and corresponding $LBP_{8,1}$ images (bottom).

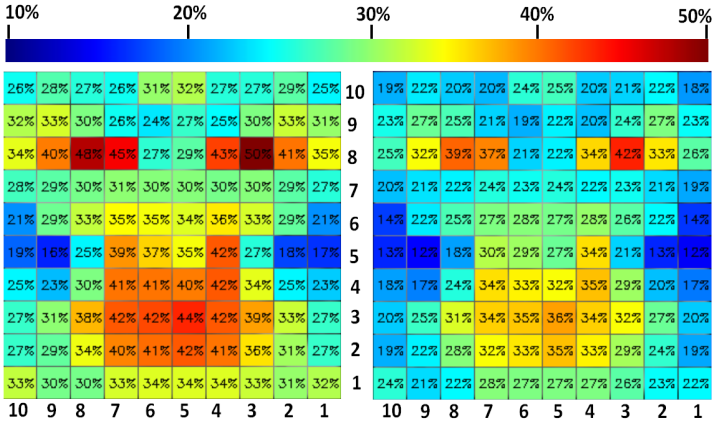


Figure 7. Color maps showing the percentage of retained energy from PCA in each region for gender (left; $\tau_e = 0.98$) and age (right; $\tau_e = 0.97$). Notice the high variance regions around the eyes and mouth.

RBF constants γ and C_p using a 10-fold cross-validation method to avoid the under or over-fitting in training stage. However, in a multi-class problem ($C > 2$) with disproportionate class sizes, the classifier must be balanced using a dedicated weight for each class. For instance, in age classification the penalty cost of a smaller dataset (*e.g.*, senior) should be decreased to counterbalance and diminish the influence of a larger dataset (*e.g.*, adult). After training, the resulting support vectors are of dimension $\mathbb{R}^{1\times(J.(C-1))}$ each, where C is the class size. We model the multi-class age classifier as a binary classification problem using one vs. one comparison amongst all classes, and a max-wins voting scheme to determine the age group.

Furthermore, we generalize the work in [25] to improve the performance on embedded system using a demography-based discriminative model for classification. As shown in Figure 5, we build a tree that discriminates the classification of gender based on the recognized ethnicity (n groups), and age (m groups) based on the recognized gender, using n separate classifiers for gender, and $2n$ separate classifiers for age recognition. The rationale behind this method roots in the differences of facial structures among different races and genders. For instance, usually middle-aged females and males do not show the same facial aging signs due to better skin-care in females. Or, different cranial structures or skin colors among races may impact the results. Thus, discrimination based on the parent stage within this tree can effectively improve the recognition rate.

On the other hand, video-based classification is more challenging than still-image-based techniques, since still-to-still classification in video sequences is an *ill-posed* problem [26]. In this case, regardless of the robustness of the classifiers, the transient variations in head-pose, facial expressions, or improper photometric conditions can cause misclassification in each frame of the video. To stabilize the results, a solution is to employ a majority voting scheme to vote for the best decisions across multiple video frames. We have integrated this *temporal voting* technique in our real-time architecture to effectively increase the confidence and reliability of decisions. Also, a face tracker can accelerate and stabilize the recognition process by continuously preserving the best classification results until the tracked face is lost. As presented in [27], detecting the best quality face images among the frames of a video sequence is another viable strategy to feed the real-time classifiers with only high quality face images, and ignore the non-informative video frames.

3. Experimental Setup

In this work, our embedded benchmarking platform was a non-real time Android system running on a multi-core 1.7 GHz Snapdragon 600 (ARMv7) SoC, with 2 GB of RAM and camera resolution 720×1280 pixels. We have implemented our framework in C++, and used Java Native Interface (JNI) to connect with Android system. Notably, several standard routines from OpenCV [28] have been integrated into our framework for face detection, photometric corrections, and SVM training. Also, a self-contained and portable binary file format is designed which includes all the parameters, support vectors, and the segmental projection matrices W_j^{EDA} and μ_j for J segments (Section 2.4). The floating-point values have single-precision for support vectors and double-precision for projection matrices.

For our embedded system we have created a training data file for gender, and two separate training data files for discriminative age recognition based on the subject's gender (Figure 5). Our age classifier categorizes four age groups of: 0-19, 20-36, 37-65, and 66+ years old. Considering the same notations used in previous sections, we begin by aligning the detected face and cropping it to size $L_o = 100$

Table 1. Databases and the number of images used for **training**

Training Database	#Images	Controlled	#Subjects	Gender		Age Group (male+female)			
				Male	Female	0 - 19	20 - 36	37 - 65	66+
FERET (fa) [5]	*1,762*	*Yes*	*1,010*	*0*	*0*	*0*	*489+357*	*270+101*	*20+0*
MORPH [29]	*55,134*	*Yes*	*13,000*	*790*	*470*	*1590+800*	*1111+1332*	*850+875*	*15+0*
Gallagher [10]	*5,080*	*No*	*28,231*	*7,350*	*7,350*	*1410+1350*	*4000+3911*	*1650+1800*	*307+312*
		Total		***8140***	***7820***	***3000+2150***	***5600+5600***	***2770+2776***	***342+312***

Table 2. Databases and the number of images used for **evaluation**

Evaluation Database	#Images	Controlled	#Subjects	Gender		Age Group (male+female)			
				Male	Female	0 - 19	20 - 36	37 - 65	66+
FERET (fb) [5]	*1,518*	*Yes*	*1,009*	*840*	*490*	*0*	*0*	*0*	*0*
Adience [30]	*26,580*	*No*	*2,284*	*3948*	*5060*	*1608+2294*	*1330+1724*	*921+1008*	*56+78*
BioID [31]	*1,521*	*Yes*	*23*	*467*	*341*	*0*	*0*	*0*	*0*
PAL [32]	*575*	*Yes*	*575*	*200*	*315*	*0*	*0*	*0*	*0*
		Total		***5455***	***6206***	***1608+2294***	***1330+1724***	***921+1008***	***56+78***

pixels with the left eye offset at $\Omega_o = \frac{L_o}{4}$ (Section 2.1). Next, the 100×100 aligned image is photometrically corrected utilizing our Filtered PS method (Sections 2.3). As the samples in Figure 6 show, this method along with uniform LBP can effectively reduce the effects of illusion of sex (Figure 2), difference in skin colors, facial cosmetics and lighting conditions while preserving facial wrinkles for age classification. In order to compensate for face localization errors, each feature segment Q_j is composed of five different radii ($R = 5$) of uniform ($L = 59$, if $P = 8$) multi-scale LBP histograms. According to our experiments, greater radii ($R > 5$) in uniform LBP could not improve the results further. Similar to Figure 4, the resulting LBP images are partitioned into 10×10 non-overlapping regions ($J = 100$) to extract the feature vector $\mathrm{Q} \in \mathbb{R}^{100\times(59*5)}$ for each sample.

For eigenvector selection in our segmental dimensionality reduction approach, we have obtained the energy threshold values τ_e (Table 3), experimentally. The color maps in Figure 7, illustrate the percentage of retained eigenvectors in each segment Q_j for age and gender classifiers. Matching the regions in the color maps and the LBP image of Figure 4, the importance of discriminative regions around the eyes and mouth is evident. Thereby, the effects of eyeglasses and facial expressions can be minimized. Furthermore, to improve the numerical stability in discriminant analysis we chose the regularization constant $\gamma = 0.01$ to avoid near-zero eigenvalues (Section 2.4). Tuning the constant γ with other values did not affect the results significantly. Table 3 lists the configuration of our classifiers such as the total number of training images, values for the threshold τ_e, and the RBF parameters. To balance the age training set, the class weights were adjusted experimentally, based on the size of each class and their influence on other classes.

Table 3. Configuration of the age and gender classifiers

Classifier	#Classes	#Training Images	PCA	RBF	
			τ_e	γ	C_p
Gender	*2*	*15,960*	*0.98*	*1.0125*	*2.5*
Age (M)	*4*	*11,712*	*0.97*	*1.0125*	*2.5*
Age (F)	*4*	*10,838*	*0.97*	*1.5187*	*2.5*

In order to evaluate these classifiers, a variety of face databases have been used as a *cross-database* benchmark for training and testing stages. As demonstrated in [7], the single-database evaluations in many researches are optimistically biased due to disproportionate diversity of races and ages, or specific lighting or head pose conditions in each database. Hence, we have trained our classifiers using a combination of selected face images from the databases listed in Table 1, and evaluated the same classifiers on a different set of databases in Table 2. Except the in-the-wild face images of Gallagher [10] and Adience [30] databases, the rest are captured in controlled lighting and head pose conditions. From Adience database, even though we have used only near-frontal version (13,649 images with $\pm 5^\circ$ yaw angle), the evaluation on this unconstrained dataset is still very challenging. Eidinger *et al*. [30] demonstrated that the difficulty level of this dataset is more than Gallagher dataset. Moreover, unlike some researches [6] that have performed evaluation on manually aligned and normalized images, we have evaluated the classifiers using our full recognition pipeline; from face detection to age and gender recognition. Therefore, our evaluation results closely reflect the real-world conditions.

4. Results and Discussion

In this section, we present the evaluation results as well as the memory and computation requirements on the embedded system. In spite of the memory-efficient and real-time performance of our method, the success rates are closely comparable with other state of the art but resource-demanding approaches. Although, the classification param-

Table 4. Age recognition rates per age group and gender (our *MSLBP+EDA+SVM* method vs. the state-of-the-art classifier). ***Note:*** only the total success rate is available for the cited paper (**#**: No. of images used). See Table 2 for no. images we used for evaluation.

Database	**Classifier** (*:embedded system)	**0 - 19**		**20 - 36**		**37 - 65**		**66+**		**Total**	
		F	**M**	**F**	**M**	**F**	**M**	**F**	**M**	**F**	**M**
Adience	***MSLBP+EDA+SVM****	***82.74%***	***93.03%***	***85.56%***	***83.53%***	***75.79%***	***75.35%***	***80.47%***	***83.59%***	***82.27%***	***85.48%***
	Dropout-SVM[30]	*#2989*	*#2487*	*#1692*	*#1602*	*#1027*	*#1148*	*#309*	*#272*	*80.7%*	

Table 5. Gender recognition rates (our *MSLBP+EDA+SVM* method vs. the state-of-the-art classifiers). ***Note:*** only the total success rate is available for the cited papers (**#**: No. of images used). See Table 2 for no. images we used for evaluation.

Database	**Classifier** (*:embedded system)	**Female**	**Male**	**Total**
BioID	***MSLBP+EDA+SVM****	***92.08%***	***98.50%***	***95.79%***
	*SHORE[34]**	*N/A*		*94.3%*
FERET	***MSLBP+EDA+SVM****	***96.12%***	***94.64%***	***95.19%***
	LUT Adaboost[13]	*#450*	*#450*	*93.33%*
	SVM+RBF[7]	*#403*	*#591*	*93.95%*
PAL	***MSLBP+EDA+SVM****	***91.43%***	***90.50%***	***91.07%***
	Adaboost[6, 7]	*#357*	*#219*	*87.24%*
	SVM+RBF[7]	*#357*	*#219*	*89.81%*
Adience	***MSLBP+EDA+SVM****	***90.77%***	***65.93%***	***79.88%***
	Dropout-SVM[30]	*#6455*	*#5824*	*75.8%*

eters can be tuned to achieve a high success rate for a specific database, it may fail to generalize the success on other databases. Some of such non-generic parameterization include: retaining eigenvectors selectively per database [7], existence of multiple same identity subjects in evaluation [4], or targeted and very low number of evaluation samples [33]. In contrast, we aim to evaluate our classifiers with the same configurations on every database. Table 5 shows the recognition rates obtained from our experiments for gender classification on databases mentioned in Table 2, and the comparisons to some existing robust classifiers.

According to our observations, the reason for lower gender recognition rate in male group (*65.93%*) of Adience database can be attributed to the existence of numerous children of under 6 years old who are very similar in appearance to females. Also, the low gender recognition rate on PAL database, confirms the influence of ethnicity on demographics classification. Provided that our training set is mostly consisted of White subjects (Table 1), the gender (or age) classifier may fail for some African subjects in this database due to different facial structures and features. Likewise, the same conditions may apply for other missing races in the training set. Exploiting the demographics discriminative classification strategy (Section 2.5), the classifier can better generalize on faces of different races.

As Table 4 shows, our evaluation results for age classification on Adience dataset outperform the results of the state-of-the-art dropout-SVM method of Eidinger *et al*. [30]. The improved accuracy can be attributed to the utilization of our illumination normalization technique, multi-scale face image representation, demography-based age classification, and non-linear SVM classification. Nevertheless, in Adience database the existence of numerous faces with masks, makeup, occlusions, and severe distortions, increases the classification errors, considerably. Particularly, in contrast to males of this dataset, many *15-19* years old females are misclassified due to high resemblance to the *20-36* age group. We believe the lower intensity of facial aging signs due to cosmetics and skin-care in females may contribute to these errors. Also, the senior age group in Gallagher (training database) starts from *66* years old, but in Adience (evaluation database) from *60* years old. This discrepancy and confusion could be the reason for lower success rates in our *37-65* and *66+* age groups.

4.1. Resource Requirements for Embedding

Technically, the SVM classifier with RBF kernel is accurate but in addition to large memory requirements, it cannot perform in real-time using a large and high-dimensional training set. For this purpose, our enhanced segmental dimensionality reduction approach is designed to supply the SVM classifier with a low-dimensional enhanced feature vector which reduces the memory and computational requirements on embedded system, remarkably. In addition, our demography-based discriminative classification model (Section 2.5) can accelerate the classification process by splitting a large training set into several smaller training sets each of which are dedicated to a specific group of gender or ethnicity. In this case, since in training stage we only include a very limited number of samples per group of training sets (*e.g*., Asian Females), then much fewer support vectors are generated for each group. Consequently, the number of computations for similarity measurement (query image vs. training data) will be reduced in testing stage.

Table 7 lists the approximate computation time of different stages of our framework using the experimental platform mentioned in Section 3. As can be seen in this table, most of the computation time for face alignment stage is spent on landmark detection. In this system, we have used the OpenCV's detection-based face tracker [28] which runs on a separate thread and we do not take its computation time into account. This face tracker searches the *whole* image only at specific intervals, and otherwise limits its searching scope to the neighborhood of the previously detected faces in each video frame. Therefore, its fast performance is less dependent on the dimensions of input image. For il-

lumination normalization (Section 2.3), the exact bilateral filters are computation-intensive, but there exist several fast approximation algorithms [18] that are embedded-friendly and can perform in real-time. On this platform, our experiments demonstrate a performance of *15* to *20* frames per second depending on the input frame rate, on-screen display parameters and, more importantly, the status of face tracker. In the latter case, the last recognition results are preserved, and it is not required to re-perform the classification until the tracked face is lost.

On the other hand, in terms of space complexity, both volatile and non-volatile memory requirements are minimized. Originally, the OpenCV's SVM trainer stores the support vectors in a very large human/machine readable file format (YAML) that is too bulky to be stored on embedded architectures. As Table 6 shows, our self-contained file format along with low dimensional training data is appropriate for most embedded architectures due to its high compression ratio of up to *99.5%*. Normally, without dimensionality reduction (*i.e.*, compression) a regular multi-scale LBP face representation with an SVM+RBF classifier would need a training data (single-precision floating-point) of dimension $\mathbb{R}^{V\times(R.L.J)}$, where V denotes the number of support vectors. However, utilizing our enhance dimensionality reduction technique, the dimension is reduced to $\mathbb{R}^{V\times(J.(C-1))}$ along with a small overhead to store the EDA transformation matrix $\mathrm{W}_j^{EDA} \in \mathbb{R}^{(LR)\times(C-1)}$ and the mean of all samples $\mu_j \in \mathbb{R}^{1\times(L.R)}$ for J segments. Based on these dimensions, we formulate the uncompressed training data size s_u (Equations 8), and the compressed training data size s_c (Equation 9), as follows:

$$s_u = V \times R \times L \times J \times E \tag{8}$$

$$s_c = \left(V\left(C-1\right) + 2LR\left(C-1\right) + 2LR\right)\left(JE\right) \tag{9}$$

where E denotes the number of bytes in floating-point type.

For instance, for gender classifier in Table 6, if $V = 5978$ support vectors, $C = 2$ classes, $R = 5$ radii, $L = 59$ bins, $J = 100$ regions, and $E = 4$ bytes floating-point, then the training data of size $s_u = 5978 \times 5 \times 59 \times 100 \times 4 =$ 672 MB, is compressed to size $s_c = (5978 \times 1 + 2 \times 59 \times 5 \times 1 + 2 \times 59 \times 5)(100 \times 4) =$ 2.8 MB (meta-data included). Although we have fewer samples for age classifier, its training data (*i.e.*, support vectors) is larger than gender classifier due to higher number of classes and larger value for RBF parameter γ.

4.2. Limitations

The most limiting factor in LDA-based classifiers is the sparsity of training samples in high-dimensional LDA subspace which can lead to overfitting. To increase the generalization capability of our approach, the number of training samples must be much larger than the number of dimensions. If the available samples are too few, utilizing PCA before LDA to reduce the dimensionality may overly discard the useful texture information. Another limitation in our current system is the lack of an ethnicity classifier which can enable us to separate the age and gender classifiers based on the subject's ethnicity, so that our classifiers can generalize their prediction capability to non-white races. Similar to [25], our experiments showed that mixing non-white faces into a single training set consisting of mostly White subjects, decreases the recognition rates of our classifiers (even by evaluating only on White subjects).

Table 6. Memory Requirements: Regular *MSLBP+SVM+RBF* vs. Our compressed file format

Classifier	#Support Vectors	MSLBP+ SVM+RBF	Our Format	Compression Ratio
Gender	*5,978*	*672 MB*	*2.8 MB*	***99.5%***
Age (M)	*8,085*	*909 MB*	*10.3 MB*	***98.8%***
Age (F)	*8,311*	*935 MB*	*10.5 MB*	***98.8%***

Table 7. Computational analysis per recognition stage

Landmark Detection	Face Alignment	Illumination Normalization
19.1 ms	*4.3 ms*	*15.5 ms*

	EDA Projection	SVM Classification
Gender	*5.9 ms*	*2.3 ms*
Age	*11.0 ms*	*3.5 ms*

5. Conclusion

In this paper, we have proposed a complete framework for real-time and accurate age and gender classification on embedded systems in unconstrained environments. Several improvements were presented for face alignment, illumination normalization, and feature extraction using a multi-resolution binary pattern method. To conquer the limitations of embedded systems, we introduced a segmental dimensionality reduction technique, and utilized a SVM+RBF classifier along with a discriminative demographics classification strategy to improve the performance. The low memory and computational requirements, makes our methodology a viable choice for real-time pattern recognition in embedded vision applications.

References

[1] C. B. Ng, Y. H. Tay, and B. Goi, "Vision-based human gender recognition: A survey," *CoRR*, vol. abs/1204.1611, 2012.

[2] T. Reponen, ed., *Information Technology Enabled Global Customer Service*. Hershey, PA, USA: IGI Global, 2002.

[3] K. Irick, M. DeBole, V. Narayanan, R. Sharma, H. Moon, and S. Mummareddy, "A unified streaming architecture for real time face detection and gender classification," *International Conference on Field Programmable Logic and Applications*, 2007.

[4] B. Moghaddam and M.-H. Yang, "Learning gender with support faces," *IEEE Trans Pattern Anal Machine Intell*, vol. 24, no. 5, pp. 707–711, 2002.

[5] P. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERet database and evaluation procedure for face-recognition algorithms," *Image and Vision Computing*, vol. 16, pp. 295–306, Apr 1998.

[6] S. Baluja and H. A. Rowley, "Boosting sex identification performance," *International Journal of Computer Vision*, vol. 71, pp. 111–119, Jun 2006.

[7] J. Bekios-Calfa, J. M. Buenaposada, and L. Baumela, "Revisiting linear discriminant techniques in gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, pp. 858–864, Apr 2011.

[8] E. Fazl-Ersi, M. E. Mousa-Pasandi, R. Laganiere, and M. Awad, "Age and gender recognition using informative features of various types," *2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014.

[9] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, pp. 51–59, Jan 1996.

[10] A. Gallagher and T. Chen, "Understanding images of groups of people," in *Proc. CVPR*, 2009.

[11] 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*, vol. 23, no. 6, pp. 643–660, 2001.

[12] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, pp. 1635–1650, Jun 2010.

[13] E. Mäkinen and R. Raisamo, "An experimental comparison of gender classification methods," *Pattern Recogn. Lett.*, vol. 29, pp. 1544–1556, July 2008.

[14] M. Uřičář, V. Franc, and V. Hlaváč, "Detector of facial landmarks learned by the structured output SVM," in *VISAPP '12: Proceedings of the 7th International Conference on Computer Vision Theory and Applications*, vol. 1, pp. 547–556, Feb. 2012.

[15] R. Russell, "A sex difference in facial contrast and its exaggeration by cosmetics," *Perception*, vol. 38, no. 8, pp. 1211–1219, 2009.

[16] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans Pattern Anal Machine Intell*, vol. 24, no. 7, pp. 971–987, 2002.

[17] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 2037–2041, Dec 2006.

[18] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," *Lecture Notes in Computer Science*, pp. 568–580, 2006.

[19] C.-H. Chan, J. Kittler, and K. Messer, "Multi-scale local binary pattern histograms for face recognition," *Lecture Notes in Computer Science*, pp. 809–818, 2007.

[20] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 711–720, Jul 1997.

[21] S. Shan, W. Zhang, Y. Su, X. Chen, and W. Gao, "Ensemble of piecewise fda based on spatial histograms of local (Gabor) binary patterns for face recognition," *18th International Conference on Pattern Recognition*, 2006.

[22] R. A. Johnson and D. W. Wichern, eds., *Applied Multivariate Statistical Analysis*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988.

[23] H. Park, "Fast linear discriminant analysis using qr decomposition and regularization," 2007.

[24] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," *Proceedings of the fifth annual workshop on Computational learning theory - COLT 92*, 1992.

[25] G. Guo and G. Mu, "Human age estimation: What is the influence across race and gender?," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, Jun 2010.

[26] A. Hadid and M. Pietikainen, "From still image to video-based face recognition: an experimental analysis," *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, 2004.

[27] A. Fourney and R. Laganiere, "Constructing face image logs that are both complete and concise," *Fourth Canadian Conference on Computer and Robot Vision (CRV 2007)*, pp. 488–494, May 2007.

[28] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[29] K. Ricanek and T. Tesafaye, "Morph: A longitudinal image database of normal adult age-progression," *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, 2006.

[30] E. Eidinger, R. Enbar, and T. Hassner, "Age and gender estimation of unfiltered faces," *IEEE Transactions on Information Forensics and Security*, vol. 9, pp. 2170–2179, Dec 2014.

[31] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz, "Robust face detection using the hausdorff distance," pp. 90–95, Springer, 2001.

[32] M. Minear and D. C. Park, "A lifespan database of adult facial stimuli," *Behavior Research Methods, Instruments, & Computers*, vol. 36, no. 4, pp. 630–633, 2004.

[33] A. Jain and J. Huang, "Integrating independent components and linear discriminant analysis for gender classification," *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, 2004.

[34] Fraunhofer IIS, "SHORE - Sophisticated High-speed Object Recognition Engine." `http://www.iis.fraunhofer.de/en/ff/bsy/dl/shore.html`.