A CBR System for Efficient Face Recognition Under Partial Occlusion

Daniel López-Sánchez[(✉)], Juan M. Corchado, and Angélica González Arrieta

C/ Espejo, s/n, Edificio I+D+i, BISITE Research Group, Salamanca 37007, Spain
{lope,angelica}@usal.es

Abstract. This work focuses on the design and validation of a CBR system for efficient face recognition under partial occlusion conditions. The proposed CBR system is based on a classical distance-based classification method, modified to increase its robustness to partial occlusion. This is achieved by using a novel dissimilarity function which discards features coming from occluded facial regions. In addition, we explore the integration of an efficient dimensionality reduction method into the proposed framework to reduce computational cost. We present experimental results showing that the proposed CBR system outperforms classical methods of similar computational requirements in the task of face recognition under partial occlusion.

Keywords: Face recognition · Partial occlusion · Dimensionality reduction

1 Introduction

This work focuses on the design and implementation of a Case-Based Reasoning (CBR) system for efficient face recognition, with a special focus on robust face recognition under partial occlusion conditions[1]. Although the problem of face recognition has been extensively addressed in the available literature, most state-ot-the-art proposals impose a series of constraintsthat limit their applicability in real world scenarios, where only a limited amount of computational power and training information is available.

The CBR method proposed in this paper seeks to cover the full recognition process (i.e., face detection, normalization, and identity prediction). In addition, we focused on methods which are able to work under the constraints of low computational power and little training information. As opposed to other occlusion-robust face recognition systems, the proposed CBR framework does not make any assumption about the nature of occlusion that it will have to face at test time. We also studied the possible integration of an efficient dimensionality reduction method in the proposed framework to reduce computational

---

[1] In the context of face recognition, partial occlusion refers to the situation where some parts of the faces the system must identify are covered by some artefact.

cost. The experimental results presented in this paper show that the proposed method outperforms traditional face recognition methods in the task of partially occluded face recognition.

The rest of this paper is structured as follows. Section 2 reviews some of the most relevant works in the field of face recognition, with special attention to approaches robust to face occlusion. The proposed CBR system and the different preprocessing methods are described in detail in Sect. 3. Section 4 empirically compares the proposed CBR system with some alternative classical methods, with special emphasis on partial occlusion scenarios. Finally, Sect. 5 summarizes the conclusions of this work and outlines some promising future research lines.

2   Related Work

In this section, we summarize some of the most relevant works on the topic of face recognition under partial occlusion. Ekenel [5] hold the idea that most of the accuracy loss registered by face recognition systems when dealing with partially occluded images is due to alignment errors, rather than information corruption by the occlusion. To address this problem, they proposed a method which seeks to minimize the distance between each sample in the training set and a new observation by evaluating a number of different alignment variations. As a consequence, searching for the best alignment variation requires hundreds of comparisons for each training sample. Although this method achieved notable accuracy rates, the computational cost supposes a major drawback.

Other authors [12,16] divide facial images into a number of delimited regions. After this, they seek to model those occlusion areas by using Principal Component Analysis or a Self Organized Map. Nevertheless, most occlusion-robust face recognition systems include a previous step to identification where they determine which parts of the images are affected by occlusion. Some studies used manually annotated occluded/non-occluded facial image patches to explicitly train a classifier [13]. However, this approach has the drawback of needing occluded face images during the training stage. As a consequence, if the nature of occlusion faced by the system in production is not the same as during the training stage, the accuracy of the occlusion detector might be affected.

Using color-based segmentation methods to detect occluded facial regions has also been proposed in the literature [7]. However, these methods are very sensitive to lighting conditions and assume that the occlusion is not caused by artefacts with human-skin color. More recently, several authors have tried to apply the recent advances in the field of *deep learning* to the task of face recognition. Nowadays, the state-of-the-art on one of the most widespread face recognition datasets, namely the *Labeled Faces in the Wild* (LFW), is held by a deep neural network trained by the scientists at Baidu [9]. The major drawbacks of this approach are the computational costs and the need for a large training dataset.

Finally, it is worth noting the CBR methodology has been applied in the literature to acquire emotional context about the users of recommender systems,

based on their facial expressions [10]. However, to the best of our knowledge the CBR methodology has not yet been applied to the task of facial recognition under partial occlusion.

3   Proposed Framework

In this section, we describe in detail both the proposed CBR framework and the selected pre-processing steps needed for face recognition. At test time, when the system is presented with an image that contains a human face in it, the following processing stages are executed: (1) A region of interest is determined for the human face in the image; (2) the detected face is aligned[2]; (3) the image is pose-normalized, rotating and scaling the face to a standard size and orientation; (4) the lighting conditions of the image are normalized; (5) a feature extraction method is applied; and finally (6) the proposed retrieval and reuse stages are executed to emit a prediction regarding the identity of the person in the original image (Fig. 1).
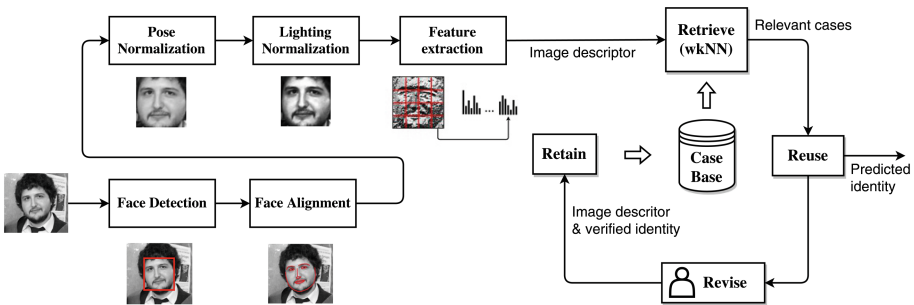

Fig. 1. Architecture of the proposed CBR framework and preprocessing steps.

3.1   Preprocessing

This section describes the successive preprocessing stages executed before the actual retrieval and reuse stages in the proposed CBR framework.

Face Detection. The face detection stage is in charge of finding a preliminary Region of Interest (ROI) for the human face present in the input image. One of the most widespread face detection methods is based on Histogram of Oriented Gradients (HOG) descriptors. This descriptor counts the number of occurrences of each gradient orientation in localized regions of the image. The face detector is then build using a linear classifier with a sliding window over the HOG descriptor of the image. For our experiments, we used the HOG face detector provided by the Dlib C++ library [8].

---

[2] In the context of face recognition, face alignment refers to the task of locating a series of facial key-points in an image, such as eyes, nose, mouth corners, etc.

Face Alignment. Face alignment consist of automatically predicting the location of a series of facial key-points in the input image. Some of the most popular methods are based on the idea of cascade regression, which provides a greater accuracy and faster processing times than classical methods. In particular, our framework leverages the face alignment algorithm proposed by V. Kazemi in 2014 [17]. Here, the author proposes using a cascade regression model where each successive level refines the alignment coordinates proposed by the previous level. In particular, the base regression models used by V. Kazemi consisted of regression-tree ensembles. For the experiments with automatic face alignment in this paper we used the pretrained model provided by Dlib C++ [8].

Pose Normalization. Once face detection and alignment have been performed, the estimated position of facial key-points in the input image is available. A pose-normalized image is then generated with these facial key-points as a basis by rotating and cropping the image to display the aligned face in its center, in a vertical pose. In addition, the resulting image is resized to a standard size.

Light Normalization. Light normalization algorithms seek to reduce the amount of intra-class variance exhibited by images from face recognition tasks with unconstrained lighting conditions. Histogram Equalization (HE) is arguably the simplest option for light normalization. This method maps the histogram of the original image $H(i)$ to a more uniform distribution. To achieve this, the so called cumulative distribution function $H'(i)$ is used:

$$H'(i) = \sum_{j=0}^{i} H(j) \tag{1}$$

Once $H'(i)$ has been computed, it is normalized to ensure that its maximum value corresponds to the maximum valid pixel value in the desired image format. Next, the following function is used to calculate pixel intensities in the resulting image:

$$equalizada(x, y) = H'(original(x, y)) \tag{2}$$

Due to its simplicity, efficiency and good performance, HE was used to normalize the lighting conditions in all experiments in this paper.

3.2 Feature Extraction: Local Binary Patterns

Using raw pixel values as features to directly train some classification algorithm is not very practical. The main reason for this being that such representation of images often contains undesired information such as noise or lighting variations. In addition, the number of pixels in images is usually too big to train a classifier efficiently. In this paper, we focus on a specific family of feature descriptors known as *Local Binary Patterns* (LBP) [14]. As described in the following sections, the localized nature of this descriptor will allow us to maintain features from occluded regions isolated from those extracted from visible parts of the face.

The LBP descriptor labels pixels in an image by considering value differences with their neighbors. This label is treated afterwards as a binary number. The use of a circular neighborhood and bilinear interpolation over non-integer pixel coordinates enables the use of this descriptor for an arbitrary neighbor number and neighborhood radio [15]. The notation $LBP_{P,R}$ is often used to refer to the LBP descriptor with P neighbors and a radio of value R. It has been proved that, using the $LBP_{8,1}$ descriptor, almost 90% of extracted labels are uniform (i.e., its binary representation contains two transitions at most) [15]. For this reason, a variant of LBP was designed where non-uniform patterns are merged together in a single label. This variant of the descriptor is known as uniform LBP ($LBP_{P,R}^u$).

Before training a classifier, the LBP representation is often refined by dividing the image in a number of blocks (arranged in a grid structure) and counting the number of concurrences of patterns in each block. After this, the corresponding histograms of each block are concatenated to form the final descriptor. This process in known as *Local binary pattern histograms* (LBPH).

3.3   Identification: Occlusion-Robust Retrieval and Reuse

The core proposal of this paper consists of a novel dissimilarity function which dynamically inhibits the use of corrupted features while retrieving the most relevant cases from the Case-Base. This section describes how this dissimilarity function is computed and its usage in the context of the proposed CBR framework.

Retrieval and Reuse. First, we introduce a method to detect partial occlusion in LBPH blocks. Conveniently, our method requires no *a priori* knowledge about the nature of occluded blocks. We define the *minimum local distance* for the histogram of an LBP block as the minimum squared Euclidean distance obtained when comparing this histogram with the LBP histograms corresponding to the same facial region in the descriptors stored in the Case-Base. Then, the only assumption made by our method is that minimum local distances of occluded blocks are usually larger than those of unoccluded blocks. To provide insight into the veracity of this assumption, we calculated the distribution of minimum local distances for occluded and unoccluded blocks in the ARFace database[3]; the resulting distributions are shown in Fig. 2. Although some overlapping exists among the two distributions, it might be possible to define a conservative threshold to discard most occluded blocks. More details on how an appropriate value for this threshold is determined can be found in Sect. 4.

Formally, the Case-Base of our framework is defined as a set of identity label y and LBPH descriptor x pairs:

$$CB = \{(y^{(i)}, x^{(i)}), i = 1, 2, \ldots, n\} \tag{3}$$

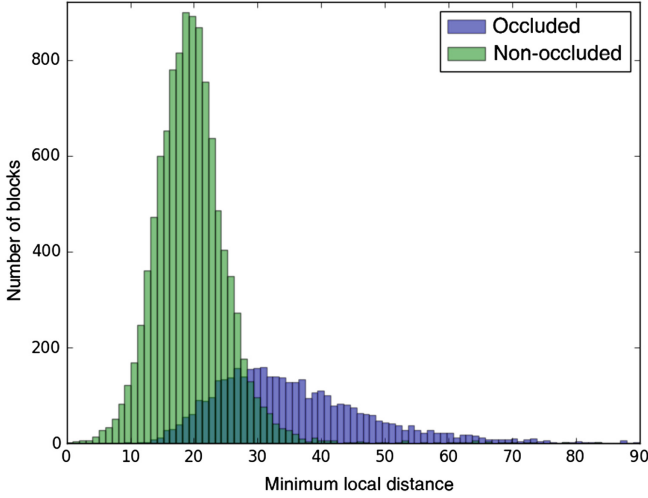[3] See Sect. 4 for details about the evaluation database.

Fig. 2. Differential distribution of minimum local distances for occluded and unoccluded blocks in the ARFace database.

When an unlabeled image I is presented to the system, it is first transformed by the successive preprocessing steps defined in the previous section. Afterwards, the LBPH descriptor $x \in \mathbb{R}^d$ of image I is generated and the retrieval stage begins. Our proposed retrieval stage begins by computing the $n \times d/p$ local distance matrix L, where p is the size of each histogram concatenated to form the LBPH descriptors. Each entry L_{ij} in this matrix corresponds to the local distance between the j-th histogram in x and the j-th histogram in the i-th descriptor in the Case-Base; formally:

$$L_{i,j} = ||(x_{p(j-1)+1}, \cdots, x_{pj}) - (x^{(i)}_{p(j-1)+1}, \cdots, x^{(i)}_{pj})||^2$$
$$\text{for} \quad i = 1, 2, \cdots, n \text{ and } j = 1, 2, \cdots, d/p \tag{4}$$

Based on this matrix and the desired *threshold* value for occlusion detection, the retrieve stage computes an occlusion mask $M \in \{0,1\}^{d/p}$ that determines which of the histograms that conform descriptor x are considered as occluded:

$$M_j = T_h(min(col_j L))$$
$$T_h(x) = \begin{cases} 1 \text{ if } x < threshold \\ 0 \text{ if } x > threshold \end{cases} \tag{5}$$

Using this occlusion mask, the retrieval stage of the proposed CBR framework finds the k most similar cases to x in the Case-Base, according to the following dissimilarity function:

$$d(x, x^{(i)}) = \sum_{j=1}^{j=d/p} M_j \cdot L_{i,j} \tag{6}$$

Intuitively, this dissimilarity function corresponds to the squared Euclidean distance between the features in x and $x^{(i)}$ that do not come from occluded facial zones (as predicted in the previous step). In other words, the proposed similarity measure dynamically inhibits the use of corrupted features while retrieving the most relevant cases from the Case-Base. Note that local distances computed in Eq. 4 are reused by the dissimilarity function. This is possible due to the fact that the squared Euclidean distance between two vectors is equal to the sum of squared Euclidean distances between segments of those vectors:

$$
\begin{aligned}
||x - y||^2 = ||z||^2 &= z_1^2 + z_2^2 + \cdots + z_i^2 + z_{i+1}^2 + \cdots + z_j^2 \\
&= ||(z_1, \cdots, z_i)||^2 + ||(z_{i+1}, \cdots, z_j)||^2 \\
&= ||(x_1 - y_1, \cdots, x_i - y_i)||^2 + ||(x_{i+1} - y_{i+1}, \cdots, x_j - y_j)||^2
\end{aligned}
\tag{7}
$$

Afterwards, the reuse stage analyses the retrieved cases and their labels to emit a prediction regarding the identity associated to the new case. To this extent, we use the weighted voting scheme proposed in [6]. First, the dissimilarities are used to compute the weight vector:

$$
w_j = \frac{1}{d(x, x^{(j)})} \text{ for } j = 1, \cdots, k
\tag{8}
$$

As explained in [6], the weight vector can be used to estimate the probability that sample x belongs to class c by:

$$
P(y = c \mid x) = \frac{\sum_{j=1}^{k} w_j \cdot I(y_j = c)}{\sum_{j=1}^{k} w_j}
\tag{9}
$$

where $I(y_j = c)$ returns a value of one if the j-th retrieved case belongs to class c, and zero otherwise. Finally, the reuse module obtains the predicted class label as follows:

$$
y = arg \max_{c \in C} P(y = c \mid x)
\tag{10}
$$

where C is the set of all possible class labels (identities).

Computational Complexity. The computational complexity of classical case-retrieval methods (i.e., nearest neighbour search) mainly depends on the method chosen to store the Case-Base. The simplest storage and search method, known as *Naive search*, stores the cases of the Case-Base without any particular order and performs a sequential search over the complete Case-Base in test time. As a consequence, the computational complexities of training and test phases are $\mathcal{O}(1)$ and $\mathcal{O}(nd + nk)$ respectively[4], where n is the number of training cases, d their dimension and k the desired number of nearest neighbours to be considered [18].

---

[4] This complexity corresponds to the version of the algorithm which computes and stores dissimilarities in a vector of dimension n. If distances are re-computed to find each nearest neighbour, the complexity is $\mathcal{O}(knd)$.

The hyperparameter k is usually considered as a constant. Hence, the complexity of test stage simplifies to $\mathcal{O}(nd)$.

Regarding the proposed method, the training stage has a constant computational cost $\mathcal{O}(1)$ as no computation is performed. For test stage, the computations defined by Eqs. 4 and 5 can be done at the cost of time $\mathcal{O}(n(\frac{d}{p} \cdot p + \frac{d}{p}))$, which simplifies to $\mathcal{O}(nd)$ given that $p > 1$. Finding the most similar cases in the Case-Base according to Eq. 6 takes $\mathcal{O}(n \cdot \frac{d}{p} + kn)$ time; which can be simplified to $\mathcal{O}(nd)$ by considering hyperparameters p and k as constants. Finally, the remaining computations which correspond to the voting process have a complexity of $\mathcal{O}(k)$. Therefore, the computational complexity of the complete test stage is $\mathcal{O}(nd) + \mathcal{O}(nd) + \mathcal{O}(k)$, which simplifies to $\mathcal{O}(nd)$ given that $n >> k$. Hence, we can conclude that the scalability of the proposed retrieval and reuse stages is equivalent to that of classical nearest neighbour search methods.

Revise and Retain. The Revise and Retain stages enable the over-time learning capabilities of the CBR methodology. In the context of the proposed CBR system, the revision should be carried out by a human expert who determines whether an image has been assigned the correct identity. The proposed method can be categorized as a *lazy learning* model, as the generalization beyond training data is delayed until a query is made to the system. As a consequence, the proposed system does not involve training any classifier or model apart from the storage of cases in the Case-Base (as opposed to other occlusion-robust face recognition approaches [12,13,16]). For this reason, retaining revised cases only involves storing their case representation into the Case-Base. In addition, this mechanism can also be applied to provide the CBR system with knowledge of previously unseen individuals, thus extending the number of possible identities predicted by the system.

3.4   Multi-scale Local Binary Pattern Histograms

Several studies have found that higher recognition accuracy rates can achieved by combining LBPH descriptors extracted form the same image at various scales [2,3]. In spite of containing some redundant information, the high-dimensional descriptors extracted in this manner are known to provide classification methods with additional information which enables higher accuracy rates. Unfortunately, the computational costs derived from using such a high-dimensional feature descriptor suppose a serious problem. Apart from that, this image descriptor is perfectly compatible with the proposed method. The only requirement is that histograms corresponding to the same image region are placed next to each other when forming the final descriptor. Then, selecting the correct value for p, the corresponding histograms for a specific face region will be treated as a single occlusion unit (i.e., a set of features which our method considers as occluded or non-occluded as a whole).

Local Dimensionality Reduction with Random Projection. This section tries to address the problem of high-dimensionality of multi-scale LBPH descriptors. In the literature, Chen *et al.* [3] proposed using an efficient dimensionality reduction algorithm to reduce the size of multi-scale LBP descriptors. However, this approach is not directly compatible with the method proposed in the previous section. The reason for this being that we need to keep features from different occlusion units (i.e., facial regions) isolated form each other, so we can later detect and inhibit features coming from occluded facial areas. Classical dimensionality reduction methods such as Principal Component Analysis and Linear Discriminant Analysis produce an output feature space were each component is a linear combination of input features, thus being incompatible with our occlusion detection method. To overcome this limitation, we propose performing dimensionality reduction at a local level. To this extent, the histograms extracted from a specific facial region (at various levels) are considered a single occlusion unit. Then, the Random Projection [1] (RP) algorithm is applied locally to each occlusion unit. As opposed to other dimensionality reduction methods, RP generates the projection matrix from a random distribution. As a consequence, the projection matrix is data-independent and cheap to build.

The main theoretical result behind RP is the Johnson-Lindenstrauss (JL) lemma. This result guarantees that a set of points in a high dimensional space can be projected to a Euclidean space of much lower dimension while approximately preserving pairwise distances between points [4]. Formally, given $0 < \epsilon < 1$, a matrix X with n samples from $\mathbb{R}^p$, and $k > 4 \cdot ln(n)/(\epsilon^2/2 - \epsilon^3/3)$ a linear function $f : \mathbb{R}^p \to \mathbb{R}^k$ exists such that:

$$(1 - \epsilon)||u - v||^2 \leq ||f(u) - f(v)||^2 \leq (1 + \epsilon)||u - v||^2 \quad \forall u, v \in X \qquad (11)$$

In particular, the map $f : \mathbb{R}^p \to \mathbb{R}^k$ can be performed by multiplying data samples by a random projection matrix R drawn from a Gaussian Distribution:

$$f(x) = \frac{1}{k} x \cdot R \qquad (12)$$

Scale 1/2 (4x4)

Scale 1 (8x8)
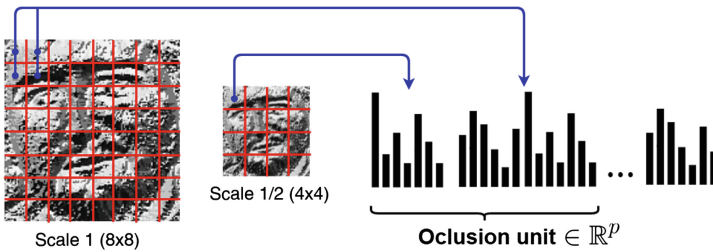
Oclusion unit $\in \mathbb{R}^p$

Fig. 3. Features are grouped together in the final descriptor according to the facial region they come from.

As previously said, in order to apply RP in the context of the proposed method, me must first ensure that histograms coming from the same face region

are placed together in the final descriptor[5] (see Fig. 3). Afterwards, we can apply the RP method locally to each occlusion unit. Formally, let $x \in \mathbb{R}^d$ be a multi-scale LBPH descriptor where each occlusion unit consists of p features, k a natural number such that $k < p$, and R a $p \times k$ random matrix whose entries have been drawn from $\mathcal{N}(0, 1)$. The reduced version of descriptor x is computed as follows:

$$x' = f((x_1, \cdots, x_p)) \ || \ f((x_{p+1}, \cdots, x_{2p})) \ || \ \cdots \ || \ f((x_{d-p+1}, \cdots, x_d)) \quad (13)$$

where $||$ denotes vector concatenation. Note that, thanks to the JL-lemma, for a sufficiently large k value the result of applying the proposed retrieval and reuse stages over reduced descriptors is approximately the same as doing it over the original high-dimensional descriptors. To prove this, it suffices to consider the different computations carried out by the proposed retrieval and reuse stages. First, the local distance matrix is computed according to Eq. 4. If we reduce both the new case x and the descriptors $x^{(i)}$ in the Case-Base as described in Eq. 13, and set hyperparameter p to the new size of occlusion units (i.e., $p = k$), Eq. 4 can be rewritten as follows:

$$L'_{i,j} = ||(x'_{k(j-1)+1}, \cdots, x'_{kj}) - (x'^{(i)}_{k(j-1)+1}, \cdots, x'^{(i)}_{kj})||^2$$
$$\text{for} \quad i = 1, 2, \cdots, n \ \text{and} \ j = 1, 2, \cdots, d'/k \quad (14)$$

where $x' \in \mathbb{R}^{d'}$. Then, applying the JL-lemma, for a sufficiently large k value we can ensure that:

$$(1 - \epsilon) \ L_{i,j} \le L'_{i,j} \le (1 + \epsilon) \ L_{i,j}$$
$$\text{for} \quad i = 1, 2, \cdots, n \ \text{and} \ j = 1, 2, \cdots, d'/k \quad (15)$$

In other words, the distortion induced in matrix L' with respect to L is bounded. The following steps of the proposed method are based on L'. Therefore, if the difference between L' and L is small enough, the proposed retrieval and reuse stages will provide the same results when executed over the reduced descriptors. Section 4 reports on several experiments where the descriptors were reduced with this approach.

4   Experimental Results

This section reports on a series of experiments carried out to assess the performance of the proposed CBR framework in the task of face recognition under partial occlusion. We evaluated the proposed system over a database of facial images with different types of occlusion and using different image descriptors. In addition, we evaluate how much accuracy is lost by using an automated face alignment method as compared to manual human annotations. In particular,

---

[5] To ease this, we always select gird sizes such that occlusion units defined as cells in the smallest grid contain an integer number of cells from the bigger grids.

the evaluation dataset is the *ARFace database* [11]. This dataset contains about 4,000 color images corresponding to 126 individuals (70 men and 56 women). The images display a frontal view of individuals' faces with different facial expressions, illumination conditions and partial occlusions. The dataset also includes annotations with the exact bounding boxes of faces inside images.

We used the images in the ARFace dataset to create several subsets for our experiments. In particular, we arranged a training set, a validation set, and several test sets with different characteristics:

- *Training set*: one image per individual (neutral, uniform lighting, first session).
- *Validation set*: almost one image per individual[6] (neutral, uniform lighting, second session).
- *Lighting test set*: almost four images per individual (neutral, illumination left/right, first and second sessions).
- *Glasses test set*: almost two images per individual (sunglasses, uniform lighting, first and second session).
- *Scarf test set*: almost two images per individual (scarf, uniform lighting, first and second session).

We evaluate the proposed method against other common classification methods used in the field of face recognition, namely Logistic Regression (LR), Support Vector Machine (SVM) and Naive Bayes (NB). In the case of the proposed method, several hyperparameters need to be adjusted. Hyperparameter p determines the size of the occlusion unit, and is fully determined by the parametrization of the LBP descriptor. The remaining hyperparameter is the *threshold* for occlusion detection. In an ideal scenario, a set of images with partial occlusion would be available to adjust this value. However, one of the goals of this work was to design a method which could operate without any information on the nature of occlusion during training time. Fortunately, it is possible to find a suitable *threshold* value with a validation set of images without occlusion, even if such validation set contains less than one image per individual. This can be achieved by following these steps:

1. The *threshold* is initialized to a sufficiently large value (for large *threshold* values, the proposed method behaves exactly like wkNN. We can use this to determine whether the *threshold* was initialized to a sufficiently large value).
2. The proposed CBR framework is trained over the *Training set* and evaluated over the *Validation set*.
3. The *threshold* value is decreased. Steps 2 and 3 are repeated until a significant loss in the accuracy is registered. This will indicate that some non-occluded blocks in the validation set have been misclassified as occluded, so the *threshold* value is set to the previous value.

The evaluation protocol for all our experiments has been the following: (1) the classifier under evaluation is trained over the training set; (2) the validation set

---

[6] Second session images are not available for all individuals in the dataset.

is used to perform hyperparameter selection; (3) the classifier, parametrized as determined in the previous step, is re-trained over the union of the training set and the validation set; (4) the trained classifier is evaluated over the different test datasets available.

Experimental Results with Automatic Face Alignment. Table 1 presents the results obtained by using the automatic face detection and alignment methods explained in Sect. 3.1. For single scale descriptors, we used $LBP_{8,2}^u$ histograms over a 8×8 grid, thus obtaining a descriptor of $3,776$ dimensions. In the case of multi-scale descriptors, we used $LBP_{8,2}^u$ histograms over 12×12 and 6×6 grids. The resulting descriptor dimension was therefore $10,620$. Finally, for our experiments with local RP, each 295-dimensional occlusion unit in the high-dimensional multi-scale descriptor was reduced to 150 features. Therefore, the complete descriptor ended up having a dimension of $5,400$ (i.e., approximately half the original dimension).

Table 1. Experimental results with automatic face alignment.

Features	Classifier	Lighting	Scarf	Glasses
$LBP_{8,2}^u$	wkNN [6]	81.4%	39.4%	30.0%
$LBP_{8,2}^u$	Proposed CBR $p = 59$; $threshold = 27$	96.2%	83.6%	50.2%
$LBP_{8,2}^u$	SVM (poly kernel)	78.1%	36.9%	25.1%
$LBP_{8,2}^u$	Logistic regression	84.8%	45.0%	23.4%
$LBP_{8,2}^u$	Naive Bayes	82.5%	43.7%	20.1%
multi-scale $LBP_{8,2}^u$	wkNN [6]	98.8%	73.3%	34.9%
Multi-scale $LBP_{8,2}^u$	Proposed CBR $p = 295$; $threshold = 17$	99.2%	89.2%	50.6%
Multi-scale $LBP_{8,2}^u$	SVM (poly kernel)	88.1%	59.6%	27.9%
Multi-scale $LBP_{8,2}^u$	Logistic regression	96.2%	75.1%	28.3%
Multi-scale $LBP_{8,2}^u$	Naive Bayes	86.29%	72.1%	37.03%
Multi-scale $LBP_{8,2}^u$ + RP	wkNN [6]	98.5%	66.5%	31.2%
Multi-scale $LBP_{8,2}^u$ + local RP (see Sect. 3.4)	Proposed CBR $p = 150$; $threshold = 100$	98.8%	90.5%	51.0%
Multi-scale $LBP_{8,2}^u$ + RP	SVM (poly kernel)	85.5%	55.3%	20.9%
Multi-scale $LBP_{8,2}^u$ + RP	Logistic regression	93.3%	69.0%	25.5%
Multi-scale $LBP_{8,2}^u$ + RP	Naive Bayes	84.0%	54.5%	27.1%

Experimental Results with Manual Face Alignment. Table 2 compiles the results obtained by using the manual face annotations provided by the authors of the ARface database. Again, for single scale descriptors we used $LBP_{8,2}^u$ histograms over a 8×8 grid, thus obtaining a descriptor of $3,776$ dimensions. In

the case of multi-scale descriptors, we used $LBP_{8,2}^u$ histograms with 12×12 and 6×6 grid sizes. Hence, the resulting descriptor dimension was $10,620$. Finally, for our experiments with local RP, each 295-dimensional occlusion unit in the high-dimensional multi-scale descriptor was reduced to 150 features. Therefore, the complete descriptor had a dimension of $5,400$.

Table 2. Experimental results with manual face alignment.

Features	Classifier	Lighting	Scarf	Glasses
$LBP_{8,2}^u$	wkNN [6]	95.5%	76.5%	69.5%
$LBP_{8,2}^u$	Proposed CBR $p = 59$; $threshold = 30$	99.5%	91.5%	83.5%
$LBP_{8,2}^u$	SVM (poly kernel)	96.5%	75.0%	61.0%
$LBP_{8,2}^u$	Logistic regression	98.5%	81.0%	68.0%
$LBP_{8,2}^u$	Naive Bayes	94.0%	76.5%	69.0%
Multi-scale $LBP_{8,2}^u$	wkNN [6]	100%	92.0%	86.0%
Multi-scale $LBP_{8,2}^u$	Proposed CBR $p = 295$; $threshold = 111$	99.5%	97.0%	92.0%
Multi-scale $LBP_{8,2}^u$	SVM (poly kernel)	100.0%	92.0%	84.5%
Multi-scale $LBP_{8,2}^u$	Logistic regression	100.0%	93.0%	89.5%
Multi-scale $LBP_{8,2}^u$	Naive Bayes	95.5%	93.5%	89.0%
Multi-scale $LBP_{8,2}^u$ + RP	wkNN [6]	100%	92.0%	86.0%
Multi-scale $LBP_{8,2}^u$ + Local RP (see Sect. 3.4)	Proposed CBR $p = 150$; $threshold = 111$	99.5%	97.0%	92.0%
Multi-scale $LBP_{8,2}^u$ + RP	SVM (poly kernel)	100.0%	92.0%	84.5%
Multi-scale $LBP_{8,2}^u$ + RP	Logistic regression	100.0%	93.0%	89.5%
Multi-scale $LBP_{8,2}^u$ + RP	Naive Bayes	95.5%	93.5%	89.0%

5   Discussion and Future Work

This work proposed a novel CBR framework for occlusion-robust face detection. The retrieval and reuse stages of the system use a modified version of the weighted k-Nearest Neighbour [6] algorithm to dynamically inhibit features from occluded face regions. This is achieved by using a novel similarity function which discards local distances imputable to occluded facial regions. As opposed to recent deep learning-based methods, the proposed system can operate in domains where only a small amount of training information is available, and does not require any specialized computing hardware to run.

Our theoretical analysis showed that the scalability of the proposed method is equivalent to that of classical Nearest Neighbour retrieval methods. In addition, we proved that the Random Projection algorithm can be applied in a local manner to reduce the dimension of multi-level LBPH descriptors, while ensuring

that the proposed retrieval and reuse stages will perform approximately as well as they do over the original high-dimensional descriptors.

Experimental results carried out over the ARFace database show that, in most cases, the proposed method outperforms classic classification algorithms when using LBPH features to identify facial images with partial occlusion. In addition, the proposed framework exhibits a better performance under uncontrolled lighting conditions.

Our experimental results also suggest that much of the accuracy loss registered when working with occluded images is imputable to automatic-alignment errors. In this regard, investigating how automatic face alignment methods can be made more robust to partial facial occlusion emerges as very interesting future research topic. In addition, we intend to evaluate the compatibility of the proposed CBR framework with other local feature descriptors rather than LBPH and other dimensionality reduction methods, and assess the effectiveness of our method on other datasets.

References

1. Achlioptas, D.: Database-friendly random projections. In: Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, pp. 274–281. ACM (2001)
2. Chan, C.-H., Kittler, J., Messer, K.: Multi-scale local binary pattern histograms for face recognition. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 809–818. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74549-5_85
3. Chen, D., Cao, X., Wen, F., Sun, J.: Blessing of dimensionality: high-dimensional feature and its efficient compression for face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3025–3032 (2013)
4. Dasgupta, S., Gupta, A.: An elementary proof of a theorem of johnson and lindenstrauss. Random Struct. Algorithms **22**(1), 60–65 (2003)
5. Ekenel, H.K.: A robust face recognition algorithm for real-world applications. Ph.D. thesis, Karlsruhe, University, Dissertation, 2009 (2009)
6. Hechenbichler, K., Schliep, K.: Weighted k-nearest-neighbor techniques and ordinal classification. Technical report, Discussion paper//Sonderforschungsbereich 386 der Ludwig-Maximilians-Universität München (2004)
7. Jia, H., Martinez, A.M.: Face recognition with occlusions in the training and testing sets. In: 8th IEEE International Conference on Automatic Face & Gesture Recognition, 2008, FG 2008, pp. 1–6. IEEE (2008)
8. King, D.E.: Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)
9. Liu, J., Deng, Y., Huang, C.: Targeting ultimate accuracy: face recognition via deep embedding. arXiv preprint arXiv:1506.07310 (2015)
10. Lopez-de-Arenosa, P., Díaz-Agudo, B., Recio-García, J.A.: CBR tagging of emotions from facial expressions. In: Lamontagne, L., Plaza, E. (eds.) ICCBR 2014. LNCS, vol. 8765, pp. 245–259. Springer, Cham (2014). doi:10.1007/978-3-319-11209-1_18
11. Martinez, A.M.: The AR face database. CVC Tech. Rep. **24** (1998)

12. Martínez, A.M.: Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. IEEE Trans. Pattern Anal. Mach. Intell. **24**(6), 748–763 (2002)
13. Min, R., Hadid, A., Dugelay, J.-L.: Improving the recognition of faces occluded by facial accessories. In: 2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011), pp. 442–447. IEEE (2011)
14. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. **29**(1), 51–59 (1996)
15. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 971–987 (2002)
16. Tan, X., Chen, S., Zhou, Z.-H., Zhang, F.: Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft k-NN ensemble. IEEE Trans. Neural Netw. **16**(4), 875–886 (2005)
17. Tzimiropoulos, G.: Project-out cascaded regression with an application to face alignment. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3659–3667. IEEE (2015)
18. Weber, R., Schek, H.-J., Blott, S.: A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. VLDB **98**, 194–205 (1998)