Investigating Nuisance Factors in Face Recognition with DCNN Representation

Claudio Ferrari, Giuseppe Lisanti, Stefano Berretti, Alberto Del Bimbo
Media Integration and Communication Center (MICC)
University of Florence - Florence, Italy
{claudio.ferrari,giuseppe.lisanti,stefano.berretti,alberto.delbimbo}@unifi.it

Abstract

Deep learning based approaches proved to be dramatically effective to address many computer vision applications, including "face recognition in the wild". It has been extensively demonstrated that methods exploiting Deep Convolutional Neural Networks (DCNN) are powerful enough to overcome to a great extent many problems that negatively affected computer vision algorithms based on hand-crafted features. These problems include variations in illumination, pose, expression and occlusion, to mention some. The DCNNs excellent discriminative power comes from the fact that they learn low- and high-level representations directly from the raw image data. Considering this, it can be assumed that the performance of a DCNN are influenced by the characteristics of the raw image data that are fed to the network. In this work, we evaluate the effect of different bounding box dimensions, alignment, positioning and data source on face recognition using DCNNs, and present a thorough evaluation on two well known, public DCNN architectures.

1. Introduction

In Computer Vision, the human face has been studied for long time either for understanding emotional states from expressions or as biometric feature for recognizing subjects' identity. Face recognition, in particular, compared to other biometric modalities is attractive since it does not require the contact with any sensor and can be performed at a distance in an uncooperative way. However, recognition based on faces suffers from several factors that can potentially impair the accuracy of the results. Many of these factors are not directly related to the natural variability of human faces due to sex, ethnicity, age. Some of them depend on variations of the face induced by facial expressions, beard, face occlusions due to hair or accessories like glasses, scarves, etc. We refer to these factors as *intrinsic*, since the variations associated to them directly affect the face surface. On the other hand, other factors that make face recognition a difficult task are due to the *extrinsic* conditions under which the face is captured. These include ambient illumination, pose, distance, resolution of the captured images, availability of single or multiple images or videos. Three-dimensional acquisitions of the face are also possible.

Most of the research work on face recognition tried to define and extract hand-crafted features capable of capturing the traits of the face that can better discriminate from subject to subject. For many years, this has been done on images acquired in cooperative contexts. The shift from cooperative to uncooperative datasets, acquired in the wild without subjects cooperation [10], contributed to substantially advance the research in this field orienting it towards more realistic solutions. Indeed, the last few years have seen the increasing success in applying deep learning based solutions to face recognition [17, 20, 22, 23]. One substantial innovation of deep convolutional neural networks (DCNNs) is the idea of letting the deep architecture to automatically discover low-level and high-level representations from labeled (or/and unlabeled) training data, which can then be used for detecting, and/or classifying the underlying patterns. However, this implies an extremely costly training phase, where millions of parameters must be optimized, thus requiring a huge number of example images. This problem can be smoothed by learning on one dataset and then reusing such learned features in different contexts using transfer learning [24] or fine tuning [29].

Though the proliferation of deep learning based solutions for face recognition, there are several aspects of their behavior that remain not completely understood or that have not been investigated at all. In addition, the effect on the final recognition accuracy of intrinsic or extrinsic factors has been evaluated only in a limited set of cases under controlled conditions [16].

In this work, we present a thorough study on the effect that different bounding boxes, alignment and positioning variations have on deep learning based face recogni-

tion. In addition, we also experiment how different data sources (still images of video frames) weigh on the effectiveness of the representations learned through DCNNs. To this end, we first identified two publicly available and effective DCNN architectures, namely, AlexNet [13] and vgg-vd16 [17]. We trained these networks on face data in the "wild" taken from [17] and tested them on the publicly available IARPA Janus Benchmark-A (IJB-A) [12] and YouTube Faces [25] datasets.

We compared the results obtained by using the images/frames included in the original dataset, with respect to the case where these images have been normalized in a preprocessing phase. In summary, the main contributions and outcomes of this work are: *(i)* a thorough experimentation on face data in the "wild" that evaluates the effect on recognition results of bounding box dimensions, alignment, positioning and data source; *(ii)* the evidence that deep architectures do not benefit from preprocessing operations that normalize input data both at train and test time.

The rest of the paper is organized as follows: in Sect. 2, we revise existing face recognition works that use deep learning; in Sect. 3, we discuss how scale, alignment, positioning and data source affect deep learning methods for face recognition; The deep learning based representation of the face and the architectures of the networks used in this work are reported in Sect. 4; A comprehensive experimentation on deep learning based face recognition is given in Sect. 5; finally, in Sect. 6, we report discussion and conclusions.

2. Related Work

The literature on face recognition has been dominated for long-time by the definition and use of hand-crafted features such as Local Binary Patterns (LBP) [1], Histogram of Gradients (HOG) [5] or Scale Invariant Feature Transform (SIFT) [8, 14]. These features were extracted from face images and subsequently used for training classifiers like SVM [21]. The trained classifiers were ultimately used to recognize the identities. In the last few years, the scenario has been drastically changed by the combined availability of increasing computational resources and of very large datasets that made possible the effective training of neural networks with deep architecture. These learning tools showed impressive recognition results in several visual tasks, including face recognition. In the following, we revise some recent works that use DCNN architectures for face recognition.

In [23], Taigman et al. proposed DeepFace, a nine-layer deep neural network architecture for face recognition. DeepFace comprised more than 120 million parameters using several locally connected layers without weight sharing, rather than the standard convolutional layers. This network was trained on an identity labeled dataset of four million facial images belonging to more than 4,000 identities. Explicit 3D face modeling was used to align the images using a piecewise affine transformation. The learned representations coupling the accurate model-based alignment with the large facial database generalized well to faces in unconstrained environments, even with a simple classifier.

In [22], Sun et al. proposed to learn a set of high-level feature representations through deep learning for face verification. These features, referred to as Deep hidden IDentity features (DeepID), were learned through multi-class face identification tasks, whilst they can be generalized to other tasks (such as verification) and new identities unseen in the training set. DeepID features were taken from the last hidden layer neuron activations of DCNN. When learned as classifiers to recognize about 10,000 face identities in the training set and configured to keep reducing the neuron numbers along the feature extraction hierarchy, these DCNNs gradually form compact identity-related features in the top layers with only a small number of hidden neurons. These features were extracted from various face regions to form complementary and over-complete representations.

The FaceNet system proposed in [20] by Schroff et al., learned a mapping from face images to a compact Euclidean space, where distances directly correspond to a measure of face similarity. Once this space is obtained, tasks such as face recognition, verification and clustering were implemented using standard techniques with FaceNet embedding as feature vectors. A DCNN was trained to directly optimize the embedding itself, rather than an intermediate bottleneck layer as in previous deep learning approaches. Triplets of roughly aligned matching / non-matching face patches generated using an online triplet mining method were used for training, with the main benefit of a better representation efficiency. State-of-the-art face recognition performance was obtained using only 128-bytes per face.

In the work of Parkhi et al. [17], a much simpler and yet effective network architecture achieving near state-of-the-art results on all popular image and video face recognition benchmarks was proposed. On the one hand, they showed how a very large scale dataset (2.6M images of over 2.6K people) can be assembled by a combination of automation and human in the loop, and discussed the trade off between data purity and time. On the other, they traversed through the complexities of deep network training and face recognition to present methods and procedures to achieve comparable state of the art results.

The work of Masi et al. in [15], addressed unconstrained face recognition in the wild focusing on the problem of extreme pose variations. As opposed to other techniques that either expect a single model to learn pose invariance through massive amounts of training data, or normalize images to a single frontal pose, this method explicitly tackled pose variation by using multiple pose specific models and

Figure 1. Examples of different bounding box dimensions: *(top)* tight bounding boxes; *(bottom)* large bounding boxes.

rendered face images. DCNNs were used to learn discriminative representations, called Pose-Aware Models (PAMs) using 500K images from the CASIA WebFace dataset [28]. In a comparative evaluation, PAMs achieved better performance than commercial products also outperforming methods that are specifically fine-tuned on the target dataset.

Unsupervised joint alignment of images has been demonstrated to improve performance on face recognition. The alignment reduces undesired variability due to factors such as pose, while only requiring weak supervision in the form of poorly aligned examples. Following this idea, Huang et al. [9] proposed Deep funneling as a novel combination of unsupervised joint alignment with unsupervised feature learning. Specifically, they incorporated deep learning into the alignment framework. In addition, the learning algorithm was modified for the restricted Boltzmann machine by incorporating a group sparsity penalty, leading to a topographic organization of the learned filters and improving subsequent alignment results. The method was applied to the LFW database. Using the aligned images produced by this unsupervised algorithm, higher accuracy in face verification was achieved compared to prior work in both unsupervised and supervised alignment.

In [16], a comprehensive study was presented that evaluates the performance of deep learning based face representation under several conditions, including the varying head pose angles, upper and lower face occlusion, changing illumination of different strengths, and misalignment due to erroneous facial feature localization. Face representations were extracted using two successful and publicly available deep learning models, namely, VggFace [17] and Lightened CNN [26]. Images acquired in controlled conditions were used in the experiments. The obtained results showed that although deep learning provides a powerful representation for face recognition, it can still benefit from preprocessing, for example, for pose and illumination normalization. In particular, from this study it emerged that if variations included in test images were not included in the dataset used to train the deep learning model, the role of preprocessing became more important. Experimental results also showed that deep learning based representation is robust to misalignment and can tolerate facial feature localization errors up to 10% of the inter-ocular distance.

3. Face Images Preprocessing for DCNN

The effectiveness of a face recognition system based on CNN architectures depends on some main aspects. First, the network architecture and learning strategy: depending on the task, different networks and learning methodologies can be more or less effective, for instance in face recognition it has been demonstrated that deeper architectures obtain better results [17]. Second, the image content: the effect that variations in illumination, pose, expression, resolution and others have on the final performance is a crucial aspect that indeed has been extensively studied in controlled conditions [16]. Third, the data preprocessing: this includes, first of all, the detection and the clipping of the interested area, *i.e.*, the face, the compensation of nuisances such as in-plane or out-of-plane rotations, misalignments and scale differences. Finally, the source of the data, *i.e.*, whether video frames or still images are considered.

The goal of this work is to evaluate the impact that different factors have on the performance of a face recognition system based on CNN representation, rather than the image content itself. To this aim, we consider the following aspects: *(i)* bounding box dimension; *(ii)* alignment and *(iii)* positioning.

3.1. Bounding Boxes Dimension

The dimension of the bounding box that contains the face is relevant inasmuch as it works as a tradeoff between the amount of useful information, *i.e.*, the face and non-useful information, *i.e.*, background that will be fed to the network. Tighter bounding boxes will reduce the amount of background included, but, on the other hand, will eventually reduce the amount of facial information and vice versa. Since many different face detection algorithms exist, it can be beneficial to understand how these differences impact on the representation obtained through the CNN. Two different bounding box sizes have been considered during both train and test:

- *Tight*: the dimension of this bounding box is the one

returned by most of the available face detectors. It considers a square that goes from the chin to just above the eyebrows. The amount of background is minimized. See examples in Fig. 1 (top row);

- *Large*: this bounding box is taken so as to include the whole head. The amount of background is variable depending on the head position.These bounding boxes have been obtained enlarging detectors' bounding boxes by 15%, see Fig. 1 (bottom row).

3.2. Alignment

The alignment process consists in bringing all the faces in the same relative position inside the crops so as to enhance the description semantics. Although the usefulness of the alignment step is well founded for engineered computer vision methods based on hand-crafted features, it has not been fully investigated if the effort made to perform the alignment is worth when using CNN representations. To this end, we applied two different strategies on the images used both to test and train the networks:

- *Similarity Transformation*: it is performed using the eyes position, identified by either manual annotation (if available) or exploiting a landmark detector [11]. Following a standard procedure, the image is warped so that the line connecting the eyes is horizontal and the distance between them is 100px. Their relative position inside the image is kept fixed. Some examples of aligned faces are shown in Fig. 2;

- *Frontalization*: with the term "frontalization" we refer to the process of bringing a generic face image in a frontal pose. This implies the compensation of out-of-plane rotations of the head and the rendering of a virtual frontal face image. To perform the frontalization, the method in [6, 7] has been used. It exploits the 3D information provided by a 3D Morphable Model (3DMM). Through 2D and 3D landmark correspondences, the method estimates the 3D pose of the head and fits the 3D model to the face image. It then samples and associates the face pixel values to the 3D model vertices and finally renders a frontal face image. The rendered image is pixelwise aligned by construction. Some examples are shown in Fig. 3.

3.3. Positioning

If the alignment is not applied to the images, the relative position of the face inside the bounding box can vary, with more pronounced variations for larger bounding boxes. Assuming that different face detectors can produce different outputs and that we cannot exclude detection errors, the goal here is to evaluate if and how much this behavior affects the recognition. To this aim, we consider the larger

Figure 2. Faces aligned with a similarity transformation.

Figure 3. Examples of frontalized faces.

bounding boxes (we can assume that it is always possible to enlarge a bounding box if it is too tight) and take random or fixed crops out of it. In doing so, we also have the chance to understand if there are some face parts that retain more discriminative information than others.

4. Face Representation with DCNN

We used the data collected in [17] to train two DCNN architectures, namely AlexNet [13] and vgg-vd16 [17]. Different versions of these two architectures have been trained varying the preprocessing applied to the training face images. In particular, we considered different bounding boxes dimensions and alignments, as described in Sect. 3.1 and Sect. 3.2, respectively.

These networks have been trained as face classifiers considering $N = 2,622$ unique individuals. For each individual, an average of 1000 face images have been used during training, for a total of $2,622,000$ images. The final fully-connected layer containing N linear predictors, one per identity, along with the empirical softmax log-loss are used to train the classifier.

4.1. AlexNet

The architecture of this network is made up of 7 convolutional layers, each one followed by a rectification layer (ReLU). Max pooling is applied after the first two convolutional layers and before the first fully connected layer. Three fully connected layers are present.

We trained five configurations of this architecture. As input for the training of two of these networks, we considered the original training images with two different bounding boxes dimensions (tight or large), but without alignment. Then, we trained two other configurations applying the similarity transformation described in Sect. 3.2 to both tight and

large bounding boxes. For these four configurations, augmentation based on both random flip and the choice of a random crop have been used during training.

Finally we trained a network considering the frontalized images. In this case the training set comprises about $1,800,000$ images; this is due to landmarks detection failures for the remaining $800,000$ images. Data augmentation is not applied since frontalized faces are pixel-wise aligned and thus selecting a random crop would only result in a loss of information.

4.2. Vgg-vd16

We also consider the VggFace network that has been released by the authors of [17]. This network has eight convolutional blocks each one followed by a ReLU. Max pooling is applied every two convolutional layers until layer 10, then every three. The last three blocks are Fully Connected (FC). No alignment has been applied to the face images used for training. Augmentation based on both random flip and the choice of a random crop have been used during training.

In this work we exploit the publicly available pre-trained model and, for a more thorough comparison, we also trained a Vgg-vd16 network using the frontalized images and the settings described in Sect. 4.1 for the AlexNet-frontalized.

5. Experimental Results

We evaluate the performance of the different DCNNs in a set of experiments that involve the preprocessing operation presented in Sect. 3. First, we evaluate face identification and verification accuracies both for different combinations of train and test data normalization, *i.e.*, whether alignment or frontalization are applied or not, and in function of the bounding box dimension. Regarding the latter, we also conduct an experiment aimed at finding its optimal size. Then, an evaluation of which face part carry the most valuable and discriminative information is performed. Finally, a specific protocol in which gallery and probe images are divided in terms of the data source (i.e., still images or frames) is devised, so as to figure out how much this aspect influence a DCNN recognition accuracy.

Experiments have been carried out on the recently released IARPA Janus Benchmark-A (IJB-A) [12] and the YouTube Faces (YTF) [25] datasets. Both are divided in ten splits for experimental evaluation; for each trial, we use 1 split as test set and the other 9 splits as training set.

IJB-A: Released by IARPA, this dataset is specifically designed to push the challenges of face recognition to the extreme, including face imagery coming both as still images or video frames captured under severe variations of imaging conditions, focusing on the extreme cases. The dataset comprises a total of 25800 images and video frames of 500 subjects. There are two main protocols defined: face identification (1:N) and face verification (1:1); in both the protocols the identities to be matched or retrieved are expressed by means of templates, *i.e.*, sets of images/frames of the same subject. This setting is sometimes referred in literature as *template based face recognition*. Specifically, in the identification protocol, identities in the *probe* set have to be retrieved among the ones in the *gallery* set. In the gallery, each template corresponds to a single identity while in the probe set a single identity can have more than one template.

YouTube Faces: The YTF dataset collects videos from YouTube and it is specifically designed to study the problem of face verification in videos. The dataset contains 3425 videos (the average video length is 181 frames) of 1595 subjects, and the task is to decide whether two video sequences contain the same subject.

5.1. Recognition Pipeline

In order to assess the role that different image preprocessing procedures have on the final performance, we followed a standard recognition pipeline, exploiting the trained DCNNs as feature extractors and applying the preprocessing methods described in Sect. 3 to the test images. For all the tests, the output of the last fully connected layer is used as 4096-dimensional face descriptor. The latter is extracted from the images and their horizontally flipped version; the final descriptor is obtained as the average of the two. The descriptors of the training set are used to compute a PCA projection matrix to perform dimensionality reduction on the test set. Finally, we perform the matching, though in a slightly different manner for the IJB-A and YTF; specifically, for the IJB-A, the cosine distance between each image included in each template is computed and the sum of the minimum of the distances and their average is taken as final measure. We found that including the average improves the results since it attenuates the effect of possible outliers in the templates, acting as distance between the centroids of the sets. For YTF instead, for each video sequence, the average descriptor is coarsely obtained from all the frames and used as final descriptor for the subject in the sequence. The verification is then performed by computing the cosine distance between pairs of descriptors.

5.2. Preprocessing Analysis

Results for the identification and verification protocols on the IJB-A are reported in Table 1 and 2, respectively. The second and third columns indicate the type of preprocessing; *Large* or *Tight* refer to the bounding box dimension while *Original* or *Aligned* refer to whether a similarity transformation is applied to the images or not. All the possible combinations of train and test data have been experimented and for each training data type, the best configuration is reported in bold (excluding the configurations that use the frontalized version of the images). From the results, we can first observe that there is a clear advantage in using

Table 1. Results on the IJB-A dataset using AlexNet architecture with different train and test data preprocessing methods.

			Identification 1:N				Verification 1:1	
Net	Train Data	Test Data	TAR@0.01FAR	TAR@0.001FAR	Rank@1	Rank@10	TAR@0.01FAR	TAR@0.001FAR
AlexNet	Aligned Large	Aligned Large	$\mathbf{0.873 \pm 0.012}$	$\mathbf{0.728 \pm 0.029}$	$\mathbf{0.861 \pm 0.014}$	0.967 ± 0.004	$\mathbf{0.850 \pm 0.018}$	$\mathbf{0.731 \pm 0.028}$
AlexNet	Aligned Large	Aligned Tight	0.806 ± 0.014	0.603 ± 0.022	0.797 ± 0.011	0.947 ± 0.007	0.795 ± 0.019	0.651 ± 0.031
AlexNet	Aligned Large	Original Large	0.870 ± 0.008	0.712 ± 0.018	0.857 ± 0.011	$\mathbf{0.971 \pm 0.003}$	0.845 ± 0.017	0.709 ± 0.030
AlexNet	Aligned Large	Original Tight	0.832 ± 0.008	0.638 ± 0.026	0.819 ± 0.008	0.956 ± 0.005	0.833 ± 0.020	0.693 ± 0.035
AlexNet	Original Large	Aligned Large	0.887 ± 0.010	0.738 ± 0.020	0.872 ± 0.008	0.971 ± 0.004	0.854 ± 0.018	$\mathbf{0.732 \pm 0.033}$
AlexNet	Original Large	Aligned Tight	0.825 ± 0.010	0.633 ± 0.018	0.811 ± 0.014	0.955 ± 0.008	0.807 ± 0.022	0.668 ± 0.029
AlexNet	Original Large	Original Large	$\mathbf{0.894 \pm 0.010}$	$\mathbf{0.753 \pm 0.022}$	$\mathbf{0.886 \pm 0.010}$	$\mathbf{0.977 \pm 0.003}$	$\mathbf{0.862 \pm 0.020}$	0.731 ± 0.025
AlexNet	Original Large	Original Tight	0.867 ± 0.009	0.697 ± 0.016	0.857 ± 0.007	0.968 ± 0.004	0.857 ± 0.021	0.720 ± 0.040
AlexNet	Aligned Tight	Aligned Large	0.728 ± 0.025	0.516 ± 0.025	0.724 ± 0.023	0.919 ± 0.009	0.742 ± 0.026	0.606 ± 0.037
AlexNet	Aligned Tight	Aligned Tight	$\mathbf{0.827 \pm 0.013}$	$\mathbf{0.666 \pm 0.031}$	$\mathbf{0.817 \pm 0.016}$	0.939 ± 0.006	0.808 ± 0.024	$\mathbf{0.687 \pm 0.038}$
AlexNet	Aligned Tight	Original Large	0.754 ± 0.019	0.541 ± 0.027	0.749 ± 0.017	0.932 ± 0.008	0.754 ± 0.027	0.616 ± 0.031
AlexNet	Aligned Tight	Original Tight	0.816 ± 0.013	0.632 ± 0.024	0.807 ± 0.019	$\mathbf{0.946 \pm 0.005}$	$\mathbf{0.819 \pm 0.017}$	0.682 ± 0.050
AlexNet	Original Tight	Aligned Large	0.596 ± 0.024	0.330 ± 0.023	0.582 ± 0.022	0.859 ± 0.018	0.651 ± 0.020	0.515 ± 0.025
AlexNet	Original Tight	Aligned Tight	0.717 ± 0.023	0.497 ± 0.024	0.717 ± 0.020	0.911 ± 0.011	0.731 ± 0.019	0.582 ± 0.040
AlexNet	Original Tight	Original Large	0.653 ± 0.019	0.384 ± 0.031	0.642 ± 0.022	0.896 ± 0.013	0.690 ± 0.025	0.539 ± 0.024
AlexNet	Original Tight	Original Tight	$\mathbf{0.749 \pm 0.020}$	$\mathbf{0.507 \pm 0.050}$	$\mathbf{0.750 \pm 0.021}$	$\mathbf{0.924 \pm 0.009}$	$\mathbf{0.779 \pm 0.024}$	$\mathbf{0.604 \pm 0.079}$
AlexNet	Frontalized	Frontalized	0.839 ± 0.014	0.698 ± 0.032	0.832 ± 0.019	0.952 ± 0.006	0.817 ± 0.021	0.563 ± 0.125

Table 2. Results on the IJB-A dataset using the VggFace architecture with different train and test data preprocessing methods.

			Identification 1:N				Verification 1:1	
Net	Train Data	Test Data	TAR@0.01FAR	TAR@0.001FAR	Rank@1	Rank@10	TAR@0.01FAR	TAR@0.001FAR
VggFace	-	Aligned Large	0.903 ± 0.010	0.760 ± 0.028	0.890 ± 0.011	0.975 ± 0.004	0.883 ± 0.017	0.749 ± 0.030
VggFace	-	Aligned Tight	0.880 ± 0.015	0.712 ± 0.027	0.867 ± 0.013	0.967 ± 0.006	0.853 ± 0.017	0.707 ± 0.039
VggFace	-	Original Large	$\mathbf{0.926 \pm 0.011}$	$\mathbf{0.804 \pm 0.022}$	$\mathbf{0.910 \pm 0.014}$	$\mathbf{0.983 \pm 0.003}$	$\mathbf{0.896 \pm 0.016}$	$\mathbf{0.759 \pm 0.041}$
VggFace	-	Original Tight	0.914 ± 0.011	0.746 ± 0.032	0.894 ± 0.011	0.979 ± 0.003	0.888 ± 0.017	0.735 ± 0.052
Vgg-vd-16	Frontalized	Frontalized	0.852 ± 0.010	0.725 ± 0.022	0.849 ± 0.008	0.938 ± 0.006	0.824 ± 0.021	0.574 ± 0.122

larger bounding boxes both in training and testing the networks. This suggests that the networks are able to separate between useful (face) and non useful (background) content themselves while training, taking advantage from the larger amount of available information. If larger bounding boxes are used in the training phase, however, the performance loss using tighter boxes in the test phase is evidently less than the opposite case. This is somewhat not surprising, since it is evident that the networks cannot recognize visual information unseen during the training. A more surprising fact instead is that, for each testing configuration, better results are achieved when using larger boxes with non aligned data to train the networks. This evidence suggests that the networks are able to account for and be somewhat invariant to similarity transformations. This capability is beneficial also if aligned data is being tested (note that the original VggFace architecture used in this work exploits non aligned data for training). The fact that this is not true when using tighter boxes can be ascribed to the lack of meaningful visual information. We can reasonably suppose from the experimental evidence that the available visual content is not sufficient to make the network fully extrapolate the features that carry the identity information. Finally, it is worth to stress that the consistency between training and testing data is of fundamental importance; for all the different training configurations, the best performance are obtained with testing data that is consistent with the training one.

Acknowledged that larger bounding boxes lead to improved representations, an analysis on the optimal dimension is conducted. The DCNN used in this experiment is the AlexNet architecture trained on large non aligned images. Fig. 4 reports results obtained enlarging and reducing the bounding box of a certain percentage starting from a base dimension, that is the one that precisely contains the whole head. The results evidence that the latter is the optimal dimension. One could have instead expected that, since the network has been trained on larger boxes, the performance could have benefit from an enlargement. However, we observe that, being equal the percentage, the accuracy drop is relative when enlarging the box while being more significant when reducing its dimension. This suggests us that the DCNN indeed takes advantage from all the available useful information and suffers more when that information is missing rather than when more background is included.

Table 3 reports results obtained simulating different shifts in the bounding box position that can occur due to detection errors. As shown in Fig. 5, we considered 3 cases: the first case (Fig. 5 (a)) simulates slight errors in the detection; the images are resized to 256×256 and random 224×224 crops are selected. Fig. 5 (b) and (c) instead refer to more extreme cases, where respectively only the upper or the lower halves of the face are visible. We here aim at assessing which face regions carry the most of the identity information. A similar analysis regarding the occlusion of face parts is also conducted in [16], where subjects wearing sunglasses (eyes region occlusion) and scarfs (mouth-nose region occlusion) are considered. In [16] the authors show that occlusions of the eyes region dramatically worsen the

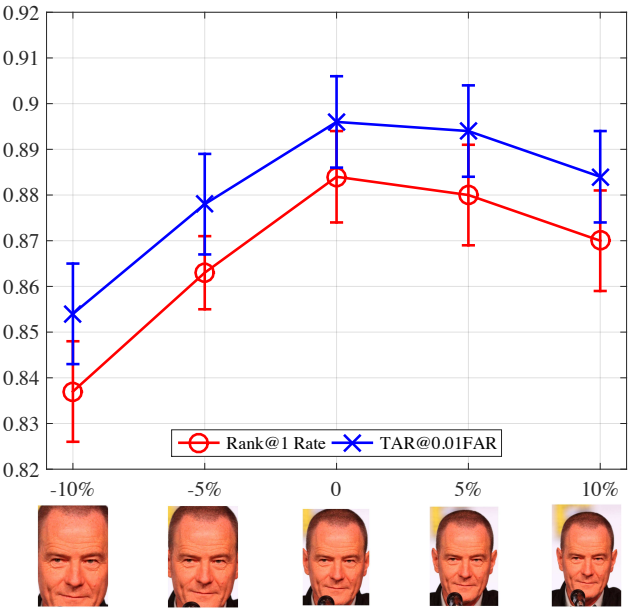
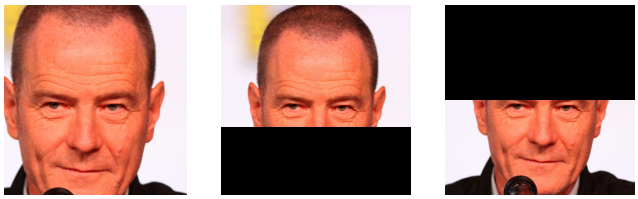


Figure 4. Recognition performance as a function of the dimension of the bounding box.

Table 3. Comparison of different bounding box positioning for the best configuration of train and test data.

	AlexNet		
	TAR@FAR 0.01	Rank 1	Rank 10
Lower-Half	0.612 ± 0.023	0.600 ± 0.018	0.881 ± 0.011
Upper-Half	0.724 ± 0.016	0.707 ± 0.015	0.924 ± 0.008
Random-Crop	0.886 ± 0.011	0.869 ± 0.011	0.974 ± 0.003
Best Configuration	$\mathbf{0.894 \pm 0.010}$	$\mathbf{0.886 \pm 0.010}$	$\mathbf{0.977 \pm 0.003}$
	VggFace		
Lower-Half	0.684 ± 0.022	0.700 ± 0.018	0.921 ± 0.011
Upper-Half	0.745 ± 0.017	0.743 ± 0.014	0.936 ± 0.005
Random-Crop	0.918 ± 0.010	0.899 ± 0.011	0.981 ± 0.003
Best Configuration	$\mathbf{0.926 \pm 0.011}$	$\mathbf{0.910 \pm 0.014}$	$\mathbf{0.983 \pm 0.003}$

recognition, while occlusions of the lower area do not influence much the results. Differently, in our experiments, we included the eyes region in both the cases (Fig. 5(b)-(c)) so as to deepen which of the two regions carries more discriminative information. Considering the asymmetric vertical position of the eyes, in order to retain approximately the same amount of visual information, we cut out a slightly smaller region for the upper half case (Fig. 5 (b)). The sizes of the crops are the 28% and 35% of the image height for the upper and lower halves respectively. Consistently with the finding of [16], Table 3 shows that removing the upper half of the face leads to a more significant drop of performance than excluding the lower half. Nonetheless, we can conclude that, since the eyes region is present in both, the eyebrows and forehead parts are of greater importance for the final representation. Randomly shifting the position of the bounding box, and so removing a small portion of the content, is instead not that crucial as data augmentation is applied in training the network.



(a) Random crop (b) Upper half (c) Lower half

Figure 5. Examples of: *(a)* random bounding box of the face; *(b)* upper part of the face visible; *(c)* lower part of the face visible.

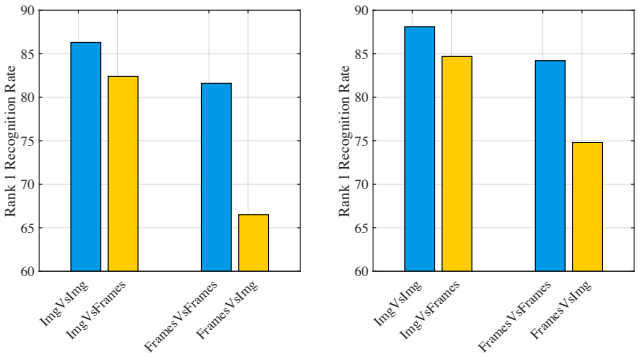


Figure 6. Rank-1 accuracy using different sources for gallery and probe; *(left)* AlexNet; *(right)* VggFace.

5.3. Data Source Analysis

As data coming from both video sequences and still images is available in the IJB-A dataset, we devised a protocol to evaluate the impact of the data source. In this protocol four setups in which gallery and probe sets contain exclusively still images or frames are considered. To this end, we select the subset of the IJB-A identities that have at least one still image and one frame. Since in the original protocol identities in the probe set can be missing in the gallery set, this selection is made only for the gallery, so as to maintain the same set across all the setups. It resulted that, for each split, 95 out of the total 112 gallery identities are retained in all the setups. For the probe set instead, images are filtered out depending on whether still images or frames are used.

In Fig. 6 is shown that actually the data source does influence the accuracy. For both the DCNN architectures a performance drop is observed when gallery and probe data come from different sources, with a much more significant loss when the gallery is composed of video frames. Being aware that generally video frames have a lower resolution than still images, we believe that the different capturing formats still lead to changes in the image content and so in the extracted representation. This suggests us that can be useful to include video frames in the training set of a DCNN.

Table 4. State of the art results on the IJB-A dataset. *Best configurations for both AlexNet and VggFace have been selected. Best results are reported in bold and second best are underlined.

	Identification 1:N				Verification 1:1	
Method	TAR@0.01FAR	TAR@0.001FAR	Rank@1	Rank@10	TAR@0.01FAR	TAR@0.001FAR
UMD (DCNN+metric) [3]	–	–	0.852 ± 0.018	0.954 ± 0.007	0.787 ± 0.043	–
UMD (DCNN$_{fusion}$) [2]	–	–	0.903 ± 0.012	0.977 ± 0.007	0.838 ± 0.042	–
PAMs [15]	–	–	0.840 ± 0.012	0.946 ± 0.007	0.826 ± 0.018	0.652 ± 0.037
Template Adaptation [4]	0.774 ± 0.050	–	0.928 ± 0.010	0.986 ± 0.003	$\underline{0.939 \pm 0.013}$	$\underline{0.836 \pm 0.027}$
TPE [19]	$\mathbf{0.932 \pm 0.010}$	0.753 ± 0.030	0.932 ± 0.010	0.977 ± 0.005	0.900 ± 0.010	0.813 ± 0.020
All-In-One CNN + TPE [18]	0.792 ± 0.020	–	$\underline{0.947 \pm 0.008}$	$\mathbf{0.988 \pm 0.003}$	0.922 ± 0.010	0.823 ± 0.020
NAN [27]	0.817 ± 0.041	–	$\mathbf{0.958 \pm 0.005}$	$\underline{0.986 \pm 0.003}$	$\mathbf{0.941 \pm 0.008}$	$\mathbf{0.881 \pm 0.011}$
AlexNet*	0.894 ± 0.010	$\underline{0.753 \pm 0.022}$	0.886 ± 0.010	0.977 ± 0.003	0.862 ± 0.020	0.731 ± 0.025
VggFace*	$\underline{0.926 \pm 0.011}$	$\mathbf{0.804 \pm 0.022}$	0.910 ± 0.014	0.983 ± 0.003	0.896 ± 0.016	0.759 ± 0.041

5.4. Comparison with State of the Art

For the sake of completeness, we compare the best configuration for both the two considered DCNNs with state of the art methods on the IJB-A and YTF datasets. Results are reported in Table 4 and Fig. 7, respectively. Results on the IJB-A show that our best configurations get very competitive results.

For what concerns the YTF dataset, we considered the original frames (without any preprocessing) to extract the DCNN descriptors. As for the bounding boxes, the provided annotations define a crop that resembles the *tight* one shown in Fig. 1. As we found that the best option is to have a large bounding box, we conducted the experiment two times, using both the original annotations and an enlarged version. The bounding boxes have been enlarged of 15% so as to approximately match the optimal dimension in the latter case. Considering that consistency between train and test data improves the results, when using the original YTF annotations, we extract feature descriptors with AlexNet trained on tight bounding boxes. The ROC curves in Fig. 7 surprisingly show that the best performance is obtained with the latter configuration, which outperforms the state of the art. One reason for this behavior can be found in the way in which the matching is performed. It is reasonable to believe that computing the average face descriptor when a certain amount of background is included in the bounding box can have a negative effect on the representation.

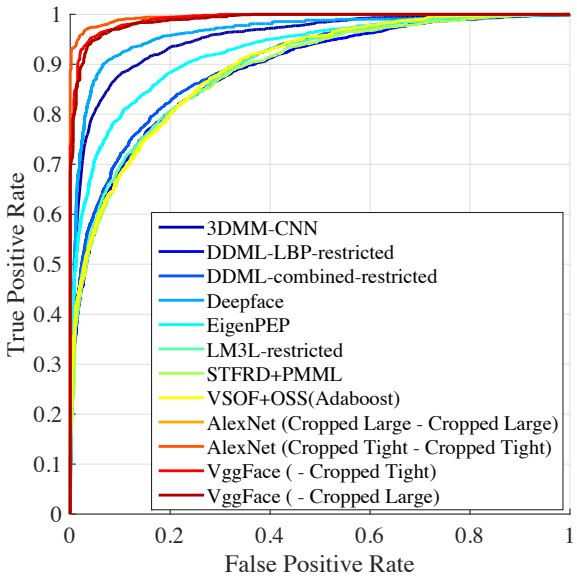


Figure 7. ROC curves on the YouTube Faces database for the trained architectures and the state of the art.

6. Conclusions

In this paper, we investigated nuisance factors that can influence face recognition performance. We focused on the images preprocessing steps, for both training and testing. From the experimental evidence we can mainly conclude that there exist a strong dependency between train and test data and that actually the image representation derived from the DCNNs does not benefit from image normalization operations. Moreover, we evidenced that the data source combinations (images or frames) has a certain impact on the final performance.

From this analysis some useful insights have also been derived that can help to lighten the effort in developing new solutions for face recognition in the wild exploiting DCNN.

Acknowledgment

This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA contract number 2014-14071600011. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon.

References

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

[2] J.-C. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep cnn features. In *IEEE Winter Conf, on Applications of Computer Vision (WACV)*, pages 1–9, 2016.

[3] J.-C. Chen, R. Ranjan, A. Kumar, C.-H. Chen, V. M. Patel, and R. Chellappa. An end-to-end system for unconstrained face verification with deep convolutional neural networks. In *IEEE Int. Conf. on Computer Vision Workshops*, pages 118–126, 2015.

[4] N. Crosswhite, J. Byrne, O. M. Parkhi, C. Stauffer, Q. Cao, and A. Zisserman. Template adaptation for face verification and identification. *arXiv preprint arXiv:1603.03958*, 2016.

[5] O. Dniz, G. Bueno, J. Salido, and F. D. la Torre. Face recognition using histograms of oriented gradients. *Pattern Recognition Letters*, 32(12):1598–1603, 2011.

[6] C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo. Dictionary learning based 3D morphable model construction for face recognition with varying expression and pose. In *Proceedings of the International Conference on 3D Vision*, pages 509–517, Lyon, France, Oct. 2015.

[7] C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo. Effective 3D based frontalization for unconstrained face recognition. In *Int. Conf. on Pattern Recognition*, 2016.

[8] C. Geng and X. Jiang. Sift features for face recognition. In *IEEE Int. Conf. on Computer Science and Information Technology*, pages 598–602, 2009.

[9] G. Huang, M. Mattar, H. Lee, and E. G. Learned-Miller. Learning to align from scratch. In *Int. Conf. on Neural Information Processing Systems (NIPS)*, pages 764–772, 2012.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, Oct. 2007.

[11] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1867–1874, 2014.

[12] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, M. Burge, and A. K. Jain. Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1931–1939, June 2015.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] I. Masi, G. Lisanti, A. D. Bagdanov, P. Pala, and A. Del Bimbo. Using 3D models to recognize 2D faces in the wild. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (Workshop on Social Intelligent Surveillance and Monitoring)*, pages 775–780, Portland, OR, USA, Jun. 2013.

[15] I. Masi, S. Rawls, G. Medioni, and P. Natarajan. Pose-aware face recognition in the wild. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4838–4846, 2016.

[16] M. Mehdipour Ghazi and H. Kemal Ekenel. A comprehensive analysis of deep learning based representation for face recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pages 34–41, 2016.

[17] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conf. (BMVC)*, volume 1, page 6, 2015.

[18] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa. An all-in-one convolutional neural network for face analysis. *arXiv preprint arXiv:1611.00851*, 2016.

[19] S. Sankaranarayanan, A. Alavi, C. D. Castillo, and R. Chellappa. Triplet probabilistic embedding for face verification and clustering. In *IEEE Int. Conf. on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8, 2016.

[20] F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, June 2015.

[21] J. Sivic, M. Everingham, and A. Zisserman. Who are you? learning person specific classifiers from video. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1145–1152, 2009.

[22] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1891–1898, June 2014.

[23] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. DeepFace: Closing the gap to human-level performance in face verification. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1708, June 2014.

[24] L. Torrey and J. Shavlik. Transfer learning. *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, 1:242, 2009.

[25] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 529–534, 2011.

[26] X. Wu, R. He, and Z. Sun. A lightened CNN for deep face representation. *CoRR*, 2015.

[27] J. Yang, P. Ren, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. *arXiv preprint arXiv:1603.05474*, 2016.

[28] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *CoRR*, arXiv:1411.7923, 2014.

[29] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.