Improving performance in small datasets via pre-trained architectures based on VGGFace and VGGFace2 datasets*

Rodolfo Simões[1] **, Bruno Kemmer**[1] **,**
Victor Ivamoto[1]**, Clodoaldo Lima**[1]

[1]University of São Paulo, São Paulo SP 03828-000, Brazil

{simoesrodolfo,bruno.kemmer,vivamoto, c.lima}@usp.br

Abstract. *Face recognition algorithms have achieved outstanding results under controlled conditions, mainly through deep learning computational techniques. However, the performance under uncontrolled conditions still needs improvement. Facial recognition systems in real-world problems often deal with uncontrolled conditions such as occlusion and pose and lighting variations, which degrade recognition performance. Despite such limitations, with enough training samples, it is still possible to reach high performance via existing deep-learning architectures. Nevertheless, the lack of training samples often results in low recognition accuracy in this domain. In this study, it has been shown that utilizing pre-trained models for the facial recognition task can enhance performance significantly in scenarios with a low number of training images available.*

Resumo. *Algoritmos de reconhecimento facial têm alcançado excelentes resultados sob condições controladas, principalmente por meio de técnicas computacionais de aprendizado profundo. No entanto, o desempenho em condições não controladas ainda precisa ser melhorado. Os sistemas de reconhecimento facial em problemas do mundo real geralmente lidam com condições não controladas, tais como, oclusões e variações de pose e iluminação, que degradam o desempenho do reconhecimento. Apesar dessas limitações, com amostras suficientes de treinamento, ainda é possível alcançar alto desempenho por meio das arquiteturas existentes de aprendizado profundo. No entanto, a falta de amostras de treinamento geralmente resulta em baixa precisão de reconhecimento nesse domínio. Neste estudo, foi demonstrado que a utilização de modelos pré-treinados para a tarefa de reconhecimento facial pode melhorar significativamente o desempenho em cenários com um baixo número de imagens de treinamento disponíveis.*

1. Introduction

The biometrics term refers to a wide range of technologies used to identify and verify a person's identity by measuring and analyzing human physiological and behavioral characteristics [Al-Raisi and Al-Khouri 2008]. Face recognition has become a leading technique for identity recognition or authentication since the beginning of biometric systems development [Hasan et al. 2021]. Facial recognition includes two different but related tasks,

*This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001

facial verification, which validates whether or not two photos belong to the same person and facial identification task that, given an image, seeks to recognize the individual.

The topic has gained notoriety and has been studied by the computer science community since the late 1980s using Principal Component Analysis (PCA) to explore the problem of facial recognition [Kaur et al. 2020]. After AlexNet became the champion of the ImageNet challenge in 2012 [Krizhevsky et al. 2012], deep learning approaches started to become quite popular in this domain [Hasan et al. 2021]. Current facial recognition methods based on convolutional neural networks have achieved remarkable progress, so some results have significantly exceeded human capacity [Duan and Zhang 2020].

However, its performance is still not enough for real-world applications [Wen et al. 2018, Adjabi et al. 2020]. Facial recognition systems in real-world problems often deal with uncontrolled conditions such as occlusion, pose, and lighting variations, which introduce intraclass variations and degrade recognition performance [Targino 2018]. Identifying facial images obtained in an unconstrained environment still poses several challenges ahead [Adjabi et al. 2020].

Despite such limitations, obtaining high performance with sufficient training samples is still possible. However, the lack of enough training samples results in low recognition accuracy in this domain [Hasan et al. 2021]. In this way, it is desired to increase the performance of deep learning methods in the facial recognition task in problems with few available data.

This research aims to evaluate through experimental analysis how face recognition accuracy can be improved in scenarios with a low number of training images per class (individual) when considering pre-trained models on large datasets. In particular, we compare the ResNet-50 and Squeeze-and-Excitation-ResNet-50 architectures pre-trained on the VGGFace2 dataset [Cao et al. 2018] and the VGG-16 architecture pre-trained on the VGGFace dataset [Parkhi et al. 2015]. Furthermore, we also investigated the three architectures referenced in pre-trained models on the Imagenet dataset [Deng et al. 2009].

The article is organized as follows. Section 2 presents the progress and difficulties in the task of facial recognition. Section 3 aims to show a brief review of the architectures and pre-trained models considered. Section 4 details the methodology used to evaluate and compare the pre-trained models' performance and presents the experimental results. Finally, section 5 presents the conclusion.

2. Literature review

The problem of facial recognition was tackled in the 1960s by computer vision researchers. The first facial recognition systems were based on geometric features (distances between predefined reference points) [Kaur et al. 2020]. After that, the topic gained notoriety in the late 1980s with the use of the Principal Component Analysis (PCA) method, in which the PCA is applied in order to find the eigenvectors that account for the most variance in the data distribution (in this case, eigenfaces) [Sirovich and Kirby 1987]. Later, in the 1990s, Linear Discriminant Analysis (LDA), also called Fisher Discriminant Analysis, was considered for using class information to maximize the variance between classes while minimizing intraclass variance [Belhumeur et al. 1997, Zhao et al. 1998].

At the same time, progress in other computer vision domains led to the devel-

opment of local feature extractors that can describe an image's texture at different locations. Feature-based approaches to face recognition consist of matching these local features across face images [Trigueros et al. 2018]. Gabor and Local Binary Patterns (LBP) have been extensively explored in this context in the 2000s [Liu and Wechsler 2002], [Ahonen et al. 2006]. Until 2012, the research community focused on separate procedures of facial recognition, such as pre-processing, feature extraction techniques, and classifiers to improve each step individually. However, the evolution was slow, and the works focused on different facial recognition issues, such as expression, lighting pose, etc. [Hasan et al. 2021].

Only after AlexNet became the ImageNet challenge champion in 2012 [Krizhevsky et al. 2012] deep learning architectures started to become popular in this domain. The introduction of convolutional neural networks (CNN) architectures offered a collection of several layers of processing neurons for feature transformation and extraction [Hasan et al. 2021], and compared to standard neural networks with similarly-sized layers, CNNs have much fewer connections and parameters and are therefore more efficient and easy to train [Krizhevsky et al. 2012]. Since then, new architectures and approaches have been proposed for the facial recognition task, and their performances have increased dramatically [Cao et al. 2018, Deng et al. 2019, Kloss et al. 2018, Kohli et al. 2018, Parkhi et al. 2015, Schroff et al. 2015, Taigman et al. 2014, Wang et al. 2018, Zheng et al. 2018].

The main advantage of deep learning methods is that they can be trained with large amounts of data to learn a facial representation robust to the training data variations. That way, instead of designing specialized features robust to different types of intraclass variations (e.g., lighting, pose, facial expression, aging, etc.), CNN can learn them from training data. On the other hand, the main disadvantage of deep learning methods is that they must be trained with extensive datasets containing enough variations to generalize to unknown samples [Trigueros et al. 2018].

To overcome the limitation of the small-sized sample, Parkhi et al. [Parkhi et al. 2015] train a CNN model on a very large scale dataset (2.6M images, over 2.6K people) and achieves comparable state-of-the-art results on the Labelled Faces in the Wild (LFW; [Huang et al. 2007]) benchmark as well as the Youtube Faces (YTF; [Wolf et al. 2011]) benchmark. The authors of [Heidari and Fouladi-Ghaleh 2020] have carried out a face recognition task on the LFW dataset by using a siamese network architecture and also applying transfer learning from a VGG network model pre-trained on ImageNet dataset to extract features from images along with Euclidean distance to calculate the similarity level. Similarly, in [Yu et al. 2016], the authors tackled the face identification task on three small target datasets via transfer learning. It was considered a VGGFACE pre-trained model [Parkhi et al. 2015] to extract features of target data.

3. Material and Methods

This section presents the deep learning architectures and pre-trained models considered in this work.

3.1. Architectures

VGG-16

VGGFace is a deep convolutional neural architecture suggested for facial recognition developed by the VGG group of Oxford University [Parkhi et al. 2015]. The authors considered three architectures based on [Simonyan and Zisserman 2014]. Here, we only consider the CNN implementation based on the VGG-Very-Deep-16 CNN architecture. The architecture comprises 11 blocks, each containing a linear operator followed by one or more non-linearities such as ReLU and max pooling. The first eight blocks are convolutional, and the last three are Fully Connected (FC). All the convolution layers are followed by a rectification layer (ReLU), the first two FC layers' outputs are 4,096 dimensional, and the last FC layer has either N = 2,622. Finally, the resulting vector is passed to a softmax layer to compute the class posterior probabilities. According to the authors, the input is a face image of size 224×224 with the average face image (computed from the training set) subtracted. The procedure is critical for the stability of the algorithm.

ResNet-50 and Squeeze-and-Excitation-ResNet-50

The VGGFace2 dataset authors [Cao et al. 2018] considered two architectures for training and experimenting with the new dataset, the ResNet-50 [He et al. 2016] and Squeeze-and-Excitation (SE)-ResNet-50 [Hu et al. 2017]. The crucial breakthrough with ResNet was that it enabled the training of deep architectures to contain hundreds of layers and still achieve outstanding performance. Before this, when deeper networks were able to start converging, a degradation problem was exposed: as the network depth increases, accuracy gets saturated and then degrades rapidly; this occurs because of the vanishing gradients problem. The authors address the degradation problem by introducing a deep residual learning framework. A residual block consists of a sequence of convolutional layers with skip connection where the input is added to the output as illustrated in Figure 1. The skip connections mitigated the vanishing gradient problem by providing an alternative path for the gradient. Therefore, instead of hoping each few stacked layers directly fit a desired underlying mapping, the approach explicitly lets these layers fit a residual mapping.

ResNet-50 is a compact variant of ResNet-152 with 50 layers. The architecture has 5 steps, each one formed with one convolution and one identity block. In the first step, we have a convolution with a kernel size of 7 * 7 and 64 different kernels, all with a stride of size 2. In the second step, a max pooling with also a stride size of 2, and there are 9 convolution layers. Then, there are 12 convolution layers in the third step. In the fourth step, there are a total of 18 convolution layers. Finally, in the fifth step, we have a total of 9 convolution layers. After that, it applies an average pool and ends with a fully connected layer containing 1000 nodes followed by the softmax function as shown in Figure 2. Furthermore, in the ResNet-50, the shortcut connections skip three layers.

The Squeeze-and-Excitation block models [Hu et al. 2017] the relationships between channels in the feature maps. The block performs channel-wise feature recalibration, strengthening meaningful features and weakening worthless ones. SE blocks fit between two layers, achieving higher performance gain at a small computational cost. The squeeze operation uses global average pooling to aggregate feature maps across their

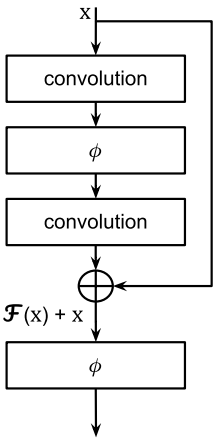

Figure 1. A residual block consists of two or more convolution layers with skip connection where the input adds to the output. ϕ is the activation function and $\bigoplus$ is element-wise sum.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 2. The Residual Networks architectures: ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152 presented in [He et al. 2016]. The first column indicates the convolution layer, the second presents the output size after the processing, and the following columns present the number of layers and kernels convolution in each architecture.

spatial dimension, and the excitation operation is a simple gating that produces a collection of weights that are applied to the feature maps. Figure 3 illustrates the architecture of the SE block.

The SE blocks can be integrated with modern architectures, such as ResNet, and improve their representational power. Thus, the authors considered it in the experiments.

3.2. Pre-trained models

To deal with the limitation of datasets with a reduced number of training images, we consider using pre-trained models on large datasets (millions of images available). Models were trained on three datasets: ImageNet [Russakovsky et al. 2015], VGGFace [Parkhi et al. 2015] e VGGFace2 [Cao et al. 2018].

The VGGFace dataset released in 2015 has 2.6 million images spanning 2,622 people, making it one of the most extensive publicly available datasets. The curated

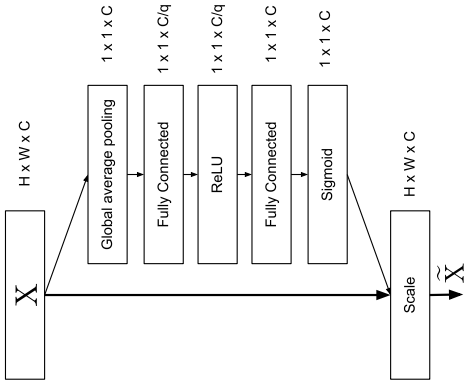

Figure 3. The squeeze and excitation block scales the feature maps in a given layer according to their importance.

version, where human annotators remove label noise, has 800,000 images with approximately 305 images per identity. The authors trained a model considering a deep convolutional neural architecture for facial recognition. They used the VGG-16 as the baseline. The model uses 224×224×3 dimensional data, the same dimension as the VGG-16. This architecture achieves very high accuracy in face recognition and can be used for any other face recognition task via transfer learning.

The VGGFace2 dataset contains 3.31 million images of 9,131 celebrities spanning a wide range of ethnicities. The dataset is roughly gender balanced, with 59.3% male, ranging from 80 to 843 images for each identity, with 362.6 images on average. The authors provided two pre-trained models on the ResNet-50 and SE-ResNet-50 architectures.

Additionally, we consider the ImageNet dataset. The ImageNet dataset comes from a large project for object recognition research. The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is a benchmark in object classification and detection across hundreds of object categories and millions of images. The challenge has been held annually since 2010, attracting the participation of researchers and institutions. Over 14 million images for over 20,000 categories were manually annotated by the project. The most used subset of ImageNet is the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012-2017, which comprises 1.28 million training images, 50,000 validation images, and 100,000 test images from 1,000 different classes. Table 1 presents the configurations of the pre-trained models considered in the experiments.

Table 1. Overview of architectures and pre-trained models considered for the experiments carried out.

Architecture	Dataset	Reference
ResNet-50	VGGFACE2	[Cao et al. 2018]
ResNet-50	ImageNet	[He et al. 2016]
SE-ResNet-50	VGGFACE2	[Cao et al. 2018]
SE-ResNet-50	ImageNet	[Hu et al. 2017]
VGG-16	VGGFACE	[Parkhi et al. 2015]
VGG-16	ImageNet	[Simonyan and Zisserman 2014]

4. Experiments

In this section, we evaluate the performance of pre-trained architectures in scenarios with few data available for training. The following experiments are developed on three configurations: (a) pre-trained architectures applied on balanced subsets of the LFW dataset; (b) pre-trained architectures applied on unbalanced subsets of the LFW dataset; (c) the performance of the pre-trained model on the VGGFace2 dataset is experimentally compared with the [Hasan et al. 2021] work that considered the pre-trained model on the VGGFace dataset. The pre-processing step, the experimental configuration, and the comparative analysis of the results will be presented and discussed in detail.

4.1. LFW experiments

For the first part of the experiments, only the Labeled Faces in the Wild (LFW) dataset will be considered. The dataset contains $13,233$ images belonging to $5,749$ people, $4,069$ individuals have only one photo, and $1,680$ have two or more photos. The term *in the Wild* demonstrates that they are images with noise; that is, they can have more than one face in the same image, and they can be in different positions and angles, which makes the task of facial recognition difficult.

LFW is a widely used dataset for face verification and identification. However, it is mostly used for the face verification problem, the images are not enough for training a state-of-the-art model for the face identification task [Wang et al. 2013]. Therefore, to deal with such limitations, the experiments in the next section were conducted to explore the trade-off between the number of available images and the performance obtained by pre-trained models on large datasets.

4.1.1. Experimental setup

The LFW dataset is unbalanced, and the vast majority of classes have a limited number of examples available for training. Figure 4 shows the distribution of images by people. With 70% of the individuals (classes) with only 1 image available, 95% with up to 5 images available, and only 0.6% with more than 30 images available. Thus, for the facial identification task experiments, we filtered from the original dataset five subsets, such that each subset contains only classes (individuals) with a minimum number of available images. The five subsets considered were 30, 25, 20, 15, and 10 images per person.

We evaluated two strategies for the experiments. The first strategy consolidated the subsets in a balanced way; each sub-dataset contains precisely the same number of examples for each person. We randomly select the number of images established for each class. The second strategy considers the problem unbalanced; it selects all images available for each person. For instance, for the experiment with the subset with 30 images, all images of all individuals that contain at least 30 images were selected (resulting in 34 individuals and 2,370 images).

For model training, we consider a batch size of 8, loss function was set to cross-entropy. The learning rate was set to 0.001, and the Stochastic Gradient Descent optimizer was utilized for the optimization process with a momentum of 0.1. The number of epochs was set to 100. For the performance evaluation of the model, the stratified cross-validation approach was considered considering five folds.

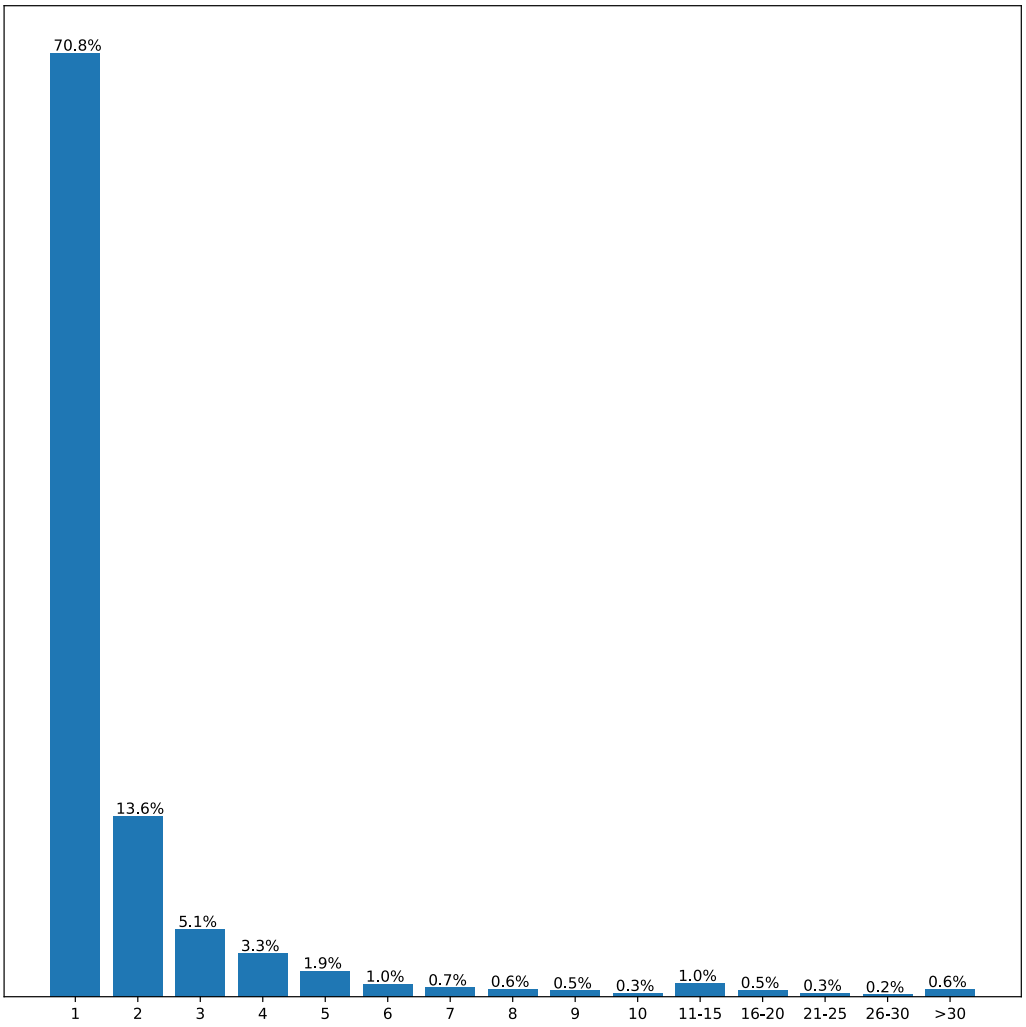

Figure 4. Distribution of total classes in the dataset by the number of images available for training. For instance, 4,096 people present in the dataset have only one image available, which represents 70.7% of the dataset available classes. Approximately 95% of classes of the LFW dataset only have up to 5 images available.

4.1.2. Pre-processing

Following the same pre-processing strategy conducted by the authors [Cao et al. 2018] for training the pre-trained model on the VGGFace2 dataset. The mean and standard deviation of the LFW dataset (RGB: mean=[0.4392, 0.3831, 0.3424] and standard deviation=[0.2684, 0.2436, 0.2348]) were considered as the normalization step. For such, the average was subtracted and divided by the standard deviation of each channel. Also, the images are resized to 224x224.

4.1.3. Result Analysis - balanced subsets LFW

In this section, we evaluate the performance of the six pre-trained models shown in Table 1 for the face identification task in five subsets of the LFW base. The results for balanced

experiments have been illustrated in Table 2 (only the experimental configurations that obtained the best results in each scenario are indicated).

Table 2. Experimental results of pre-trained models on subsets of the LFW base (balanced). The Train Data column indicates the number of examples available in training for each subset. For each architecture, we indicated only the pre-trained model that obtained the best performance in terms of average accuracy and standard deviation.

Subset	Categories	Train Data	ResNet-50		SE-ResNet-50		VGG-16	
			Pre-trained	Acc	Pre-trained	Acc	Pre-trained	Acc
30 images	34	1020	VGGFace2	**98.63**±0.57	VGGFace2	97.06±1.11	VGGFace	96.67±1.13
25 images	42	1050	VGGFace2	**98.57**±0.52	VGGFace2	96.38±1.1	VGGFace	94.76±1.08
20 images	62	1240	VGGFace2	**97.82**±0.9	VGGFace2	94.84±1.28	VGGFace	82.58±3.46
15 images	96	1440	VGGFace2	**93.82**±1.66	VGGFace2	90.56±1.52	VGGFace	74.93±0.94
10 images	158	1580	VGGFace2	**84.81**±0.49	VGGFace2	76.39±1.82	ImageNet	32.47±2.48

As shown in Table 2, there is a significant performance gain with using the pre-trained model on the VGGFace2 dataset about the pre-trained models on the VGGFace and ImageNet datasets, highlighting the performance of the pre-trained model in the ResNet-50 architecture. Furthermore, it is possible to verify that the performance of the pre-trained models in the VGGFace2 dataset does not degrade in the same proportion as the other pre-trained models according to the number of observations available per person is reduced, and the number of categories is increased. Another highlight is that there was a more significant performance degradation for the experiments with the VGG-16 architecture. Further, the pre-trained model on the ImageNet dataset performed better than VGGFace only in the scenario with10 images per individual.

4.1.4. Result Analysis - unbalanced subsets LFW

In this section, we evaluate the performance of the six pre-trained models shown in Table 1 for the face identification task in five subsets of the LFW base. The results of the unbalanced experiments are presented in Table 3.

In the same way, it is possible to verify that there is a significant gain in performance by the pre-trained models on the VGGFace2 dataset, specifically highlighting the performance of the pre-trained model in the ResNet-50 architecture. Note that there was a performance degradation of the SE-ResNet-50 architecture as the average amount of images available per person reduced and the number of categories increased. For the experiments with the VGG-16 architecture, the pre-trained model on the ImageNet dataset performed worse than the pre-trained model in VGGFace in all scenarios.

4.2. Comparative experiments

In this section, we conduct experiments to evaluate and compare the performance of the pre-trained model on the VGGFace2 dataset [Cao et al. 2018] with the pre-trained model on the VGGFace dataset presented in [Hasan et al. 2021]. The pre-processing, experimental setup and analysis of results will be presented and discussed in detail.

Table 3. Experimental results of pre-trained models applied to a subset of the LFW dataset (unbalanced). The Train Data column indicates the mean and standard deviation/number of examples available in experiments for each subset. In addition, for each architecture, it is indicated which pre-trained model obtained the best performance in terms of average accuracy and standard deviation.

Subset	Classes	Train Data	ResNet-50		SE-ResNet-50		VGG-16	
			Pre-trained	Acc	Pre-trained	Acc	Pre-trained	Acc
$\geq$ 30 images	34	69.7±89.9/2370	VGGFace2	**99.41**±0.27	VGGFace2	97.85±1.44	VGGFace	95.19±1.74
$\geq$ 25 images	42	61.6±82.6/2588	VGGFace2	**99.04**±0.24	VGGFace2	93.97±2.53	VGGFace	95.36±0.97
$\geq$ 20 images	62	48.7±70.5/3023	VGGFace2	**98.74**±0.37	VGGFace2	83.55±5.35	VGGFace	88.79±1.73
$\geq$ 15 images	96	37.4±58.6/3595	VGGFace2	**97.47**±0.83	VGGFace2	78.69±15.42	VGGFace	87.95±0.95
$\geq$ 10 images	158	27.3±47.4/4324	VGGFace2	**91.9**±1.64	VGGFace2	53.7±21.67	VGGFace	74.65±0.81

4.2.1. Dataset

For the second part of the experiments, we considered the same datasets used in [Hasan et al. 2021]. 5 celebrity faces dataset [Hossain et al. 2021] consists of 93 samples for training and 25 samples for testing across 5 classes. Georgia Tech face dataset [Mandal et al. 2007] consists of 50 classes with 500 training and 250 testing samples. KomNet dataset [Astawa et al. 2020] has a total of 1000 training and 200 testing images for 50 classes. The KomNet dataset has three subtypes, containing the same data but captured using three different mediums, i.e., digital camera (kamera), social media (sosmed), and mobile phones (hp). Moreover, all these datasets were combined and created into a single dataset with 105 classes containing 3593 training and 875 testing samples.

4.2.2. Pre-processing

Following the same strategy, each dataset's mean and standard deviation were used as a normalization step, resizing the images to 224x224. In addition, for a strict comparison with the previous work, the Multi-task Cascaded Convolutional Neural Networks (MTCNN)[1] method [Zhang et al. 2016] was considered for face detection and cropping with resizing at 224x224. The method follows three steps, the first being applied to detect a face; if there is a face, it determines the boundary box for the detected faces; and finally, it detects the landmarks (nose, mouth, and eyes).

4.2.3. Experimental setup

Following the same strategy presented in [Hasan et al. 2021]. Two experimental configurations were considered: the heuristic and non-heuristic approaches. The authors defined the heuristic term to denote prior knowledge. On this account, the approach performs a pre-training of the model for some kind of data it will encounter in the future. Further, this pre-training step makes the parameters of the network adjust themselves so that they can find the important features from the individual datasets.

For the heuristic approach, the authors performed a retraining step combining the five training datasets: 5 celebrity faces, Georgia Tech faces, and the three variants of KomNet datasets, forming a new dataset with 105 categories. After this step, the resulting

[1]Python library available on: `https://github.com/timesler/facenet-pytorch`.

pre-trained model was considered for training and testing the individual datasets. As shown in Figure 5.

For the non-heuristic approach, the pre-trained model on the VGGFace2 dataset was trained and tested individually in each dataset.

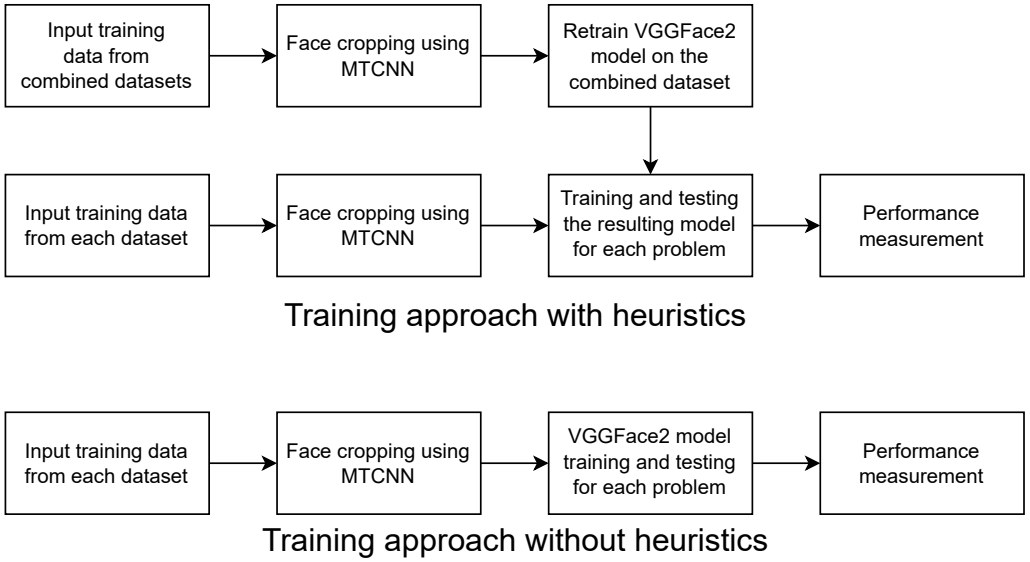

Figure 5. At the top, we have the approach with heuristics in which, first, we retrain the pre-trained model on the combined training datasets. After, the resulting model is used for training and testing each problem. At the bottom, we have the heuristic-free approach in which the pre-trained model is trained and tested individually in each problem.

In the experiments, we consider a batch size of 8, loss function was set to cross-entropy. The learning rate was set to 0.001, and the Stochastic Gradient Descent optimizer was utilized for the optimization process with a momentum of 0.1. The number of epochs was set to 100.

4.2.4. Result analysis

The authors reported for the heuristic approach that the accuracy in the KomNet Sosmed dataset is 94.41%, and for the other four datasets, the heuristic approach achieved 100% accuracy. For the non-heuristic approach was 20%, 7.6%, 89.34%, 95.5% and 96.5% to the 5 celebrity faces, Georgia tech face, KomNet Sosmed, KomNet HP and KomNet Kamera, respectively. Table 4 illustrates the results for heuristic and non-heuristic approaches. It can be noticed that the overall accuracy has improved significantly in the heuristic approach for all five datasets. Performing a comparison with the experimental results presented in the previous work is possible to notice the benefits of using the VGGFace2 pre-trained model. There was performance gain in all scenarios Non-heuristic. Besides, we achieved 100% accuracy in the scenarios with heuristics.

5. Conclusion

Facial recognition has been explored by the computer science community since the late 1980s using feature extraction methods and traditional machine learning techniques.

Table 4. Experimental results of the VGGFace2 pre-trained model. The Classes column indicates the number of classes on the problem. The Train Data and Test Data columns indicate the number of examples available for training and testing, respectively.

Dataset	Classes	Train Data	Test Data	Non-heuristic Test Acc.	Heuristic Test Acc.
5 Celebrity	5	93	25	72.0	100.0
Georgia Tech	50	500	250	82.4	100.0
KomNet (sosmed)	50	1000	200	96.45	100.0
KomNet (hp)	50	1000	200	99.0	100.0
KomNet (kamera)	50	1000	200	100.0	100.0

However, the increase in data dimensionality combined with the limitations imposed by real-life settings motivated researchers to move towards deep learning. Such approaches need enough training samples to achieve high performance. In opposite scenarios, we deal with results in low recognition accuracy in this domain. In this work, we explore the use of pre-trained models on large datasets to increase performance in problems with few data available for training. The experimental analysis verified the performance gain by using pre-trained models, especially the pre-trained model on the VGGFace2 dataset, denoting the benefit of using a pre-trained model on a dataset with 3.31 million images with a wide range of ethnicities, approximately gender-balanced, and with significant variations in pose, age, lighting, and background.

References

Adjabi, I., Ouahabi, A., Benzaoui, A., and Taleb-Ahmed, A. (2020). Past, present, and future of face recognition: A review. *Electronics*, 9(8):1188.

Ahonen, T., Hadid, A., and Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041.

Al-Raisi, A. N. and Al-Khouri, A. M. (2008). Iris recognition and the challenge of homeland and border control security in uae. *Telematics and Informatics*, 25(2):117–132.

Astawa, I. N. G. A., Putra, I. K. G. D., Sudarma, M., and Hartati, R. S. (2020). Komnet: Face image dataset from various media for face recognition. *Data in brief*, 31:105677.

Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., and Zisserman, A. (2018). Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699.

Duan, Q. and Zhang, L. (2020). Look more into occlusion: Realistic face frontalization and recognition with boostgan. *IEEE transactions on neural networks and learning systems*, 32(1):214–228.

Hasan, M. M., Hossain, M. A., Srizon, A. Y., Sayeed, A., Ahmed, M., and Haquek, M. R. (2021). Improving performance of a pre-trained resnet-50 based vggface recognition system by utilizing retraining as a heuristic step. In *2021 24th International Conference on Computer and Information Technology (ICCIT)*, pages 1–6. IEEE.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heidari, M. and Fouladi-Ghaleh, K. (2020). Using siamese networks with transfer learning for face recognition on small-samples datasets. In *2020 International Conference on Machine Vision and Image Processing (MVIP)*, pages 1–4. IEEE.

Hossain, M. I., Kabir, H., et al. (2021). An efficient way to recognize faces using mean embeddings. In *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pages 1–10. IEEE.

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. (2017). Squeeze-and-excitation networks.

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

Kaur, P., Krishan, K., Sharma, S. K., and Kanchan, T. (2020). Facial-recognition algorithms: A literature review. *Medicine, Science and the Law*, 60(2):131–139.

Kloss, R. B., Jordao, A., and Schwartz, W. R. (2018). Face verification: Strategies for employing deep models. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 258–262. IEEE.

Kohli, N., Yadav, D., and Noore, A. (2018). Face verification with disguise variations via deep disguise recognizer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 17–24.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

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

Mandal, T., Majumdar, A., and Wu, Q. J. (2007). Face recognition by curvelet based feature extraction. In *Image Analysis and Recognition: 4th International Conference, ICIAR 2007, Montreal, Canada, August 22-24, 2007. Proceedings 4*, pages 806–817. Springer.

Parkhi, O. M., Vedaldi, A., and Zisserman, A. (2015). Deep face recognition.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sirovich, L. and Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Josa a*, 4(3):519–524.

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708.

Targino, J. M. (2018). *Reconstrução de oclusões parciais em imagens de face visando o reconhecimento biométrico*. PhD thesis, Universidade de São Paulo.

Trigueros, D. S., Meng, L., and Hartnett, M. (2018). Face recognition: From traditional to deep learning methods. *arXiv preprint arXiv:1811.00116*.

Wang, H., Kang, B., and Kim, D. (2013). Pfw: A face database in the wild for studying face identification and verification in uncontrolled environment. In *2013 2nd IAPR Asian Conference on Pattern Recognition*, pages 356–360. IEEE.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., and Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274.

Wen, G., Chen, H., Cai, D., and He, X. (2018). Improving face recognition with domain adaptation. *Neurocomputing*, 287:45–51.

Wolf, L., Hassner, T., and Maoz, I. (2011). Face recognition in unconstrained videos with matched background similarity. In *CVPR 2011*, pages 529–534. IEEE.

Yu, H., Luo, Z., and Tang, Y. (2016). Transfer learning for face identification with deep face model. In *2016 7th International Conference on Cloud Computing and Big Data (CCBD)*, pages 13–18. IEEE.

Zhang, K., Zhang, Z., Li, Z., and Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503.

Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D. L., and Weng, J. (1998). Discriminant analysis of principal components for face recognition. *Face recognition: From theory to applications*, pages 73–85.

Zheng, Y., Pal, D. K., and Savvides, M. (2018). Ring loss: Convex feature normalization for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5089–5097.