Electrical and Electronics Engineering
Semester Project
Master Semester 3 - Autumn 2022

# Identity-Preserving Low-Resolution Face Recognition

*Student:* Arina Lozhkina

*Supervised by:* Yuhang Lu
Prof. Dr. Touradj Ebrahimi

January 12, 2023

Multimedia Signal Processing Group
EPFL

# Abstract

Despite significant advances in high-resolution recognition, low-resolution facial recognition remains a challenge. Many methods have been proposed for its solution, which can be divided by 2 categories: super resolution methods and resolution-invariant feature extraction. In this work the focus is on the second one.

In the project the problem of low resolution face recognition is studied including methods of solving cross-resolution and low-resolution face recognition problem, the pipeline of face recognition based on deep learning and the low resolution datasets and their evaluation protocols.

The high resolution facial recognition methods are implemented: CosFace, SphereFace and ArcFace. They are also adapted for low resolution face recognition problem using the Cross Resolution Batch training. Also the finetuning methods: Octuplet Loss and DeriveNet are implemented. The implementation of face recognition pipeline based on the existing implementation is included. All methods are tested and compared using low resolution face images.

Three low resolution datasets: LFW, SCface, QMUL-SurvFace, and their evaluation protocols are studied and implemented.

One of the modern directions is the application of adaptive margin function. The proposed methods are based on it, which adapt margins for images of different quality. The quality of the images is considered using the Laplacian operator. The proposed methods surpassed the state-of-the-art algorithm of Cross Resolution Face Recognition, AdaFace, when tested on low resolution images.

# Contents

# 1 Introduction

With the development of deep neural network technologies, the solution to the problem of face recognition in images has reached 0.9981 based on the LFW [1] testing of ArcFace [2] training loss [3]. This was facilitated by the availability of computational resources for calculations [4], extensive datasets for training [5], the general development of computer vision, which led to the formation of new architectures of models [6], loss functions [7], preprocessing algorithms [8] and image transformation [9].

However, these algorithms were trained and tested on high quality images such as LFW [1] and CelebMV1 [5]. At the same time, testing pre-trained algorithms on datasets close to real images of surveillance cameras like SCface [10] showed a significant decrease in accuracy when solving the problem of verifying faces in an image [11].

The reasons for this problem are face recognition challenges, which include low resolution, changes in facial expression, lighting, makeup, the presence of a mask, etc. For the practical application of face recognition systems, robustness of face recognition is required in images obtained in unconstrained environments [12].

This work focuses on the study, implementation and analysis of low-resolution facial recognition algorithms. In addition, within the scope of the work, new possible approaches have been developed to address the problem of low-resolution facial recognition. The second section is devoted to related work, in which the pipeline algorithm of recognition of persons based on deep learning is considered. Then there are low-resolution face recognition challenges, including low-resolution images. The following are the developed methods of solving the problem of recognition in low resolution. At the end of the review the variants of loss functions used to solve the problem of face recognition are considered. The next section focuses on the test datasets including LFW[1], QMUL-SurvFace[13], SCface [10] and their evaluation protocols. Section 4 describes the algorithms implemented within the scope, including the adaptation of high-resolution facial recognition algorithms and fine-tuning. The following are two proposed methods for solving the problem of recognition in low resolution. In the end, the values of accuracy obtained by testing all implemented algorithms in comparison with the state-of-the-art face recognition algorithm in the cross-resolution, AdaFace [14], are given.

# 2 Related Work

## 2.1 Deep Face Recognition

In recent years, deep face recognition, rather than manual extraction of features, has been an active trend [12]. In this approach, the recognition process includes 3 steps: image preprocessing, model training to form embeddings and recognition by matching vectors in feature space [12]. The pipeline is presented at 1.

The intermediate step in some modern systems is the protection against face falsification by means of 3D masks, printing, etc. [15] The result of model training for recognition is a feature vector of fixed length.

Preprocessing is needed to detect the face in the image and align it to the desired position. MTCNN Face Detection [16] is used for face detection. MTCNN applies a cascade structure of deep convolutional networks that detect face position in a coarse-to-fine manner. Each model is trained to perform a separate task: classification, bounding box regression, and anchor point localization. Between steps, non-maximal suppression (NMS) is performed to filter out strongly overlapping candidates.

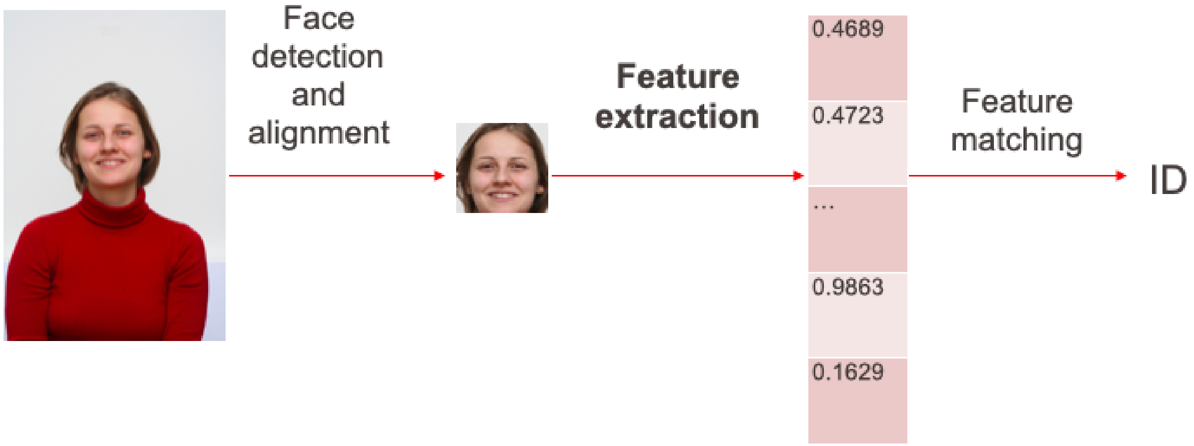

Figure 1: Face Recognition Pipeline

## 2.2 Challenges

In unconstrained environments, face recognition is complicated by possible variations associated with changes in shooting conditions and facial variations, such as expressions of emotion, pose changes, occlusions such as masks and glasses, makeup, age changes, etc. When training a model on a dataset that includes uniform face images in full-face against a neutral background, its application to images in real-world face variations may be less effective [11].

- **Poses**

  Camera placement affects the face pose of the resulting image. As a result of changes in shooting conditions, face images are no longer frontal, which leads to a deterioration in the performance of the recognition model [17]. To assess the quality of recognition during changes in the pose of the face, the protocol P dataset Multi-PIE [18], which contains images of the face when the camera is rotated in 15 degree steps, is used. Frontal images are used as a template.

- **Partial occlusion**

  Partial occlusion is a common problem that involves partial concealment of the face with glasses, mask, and other elements, and affects face recognition performance [19]. To evaluate the quality of recognition under partial occlusion, tests are performed on datasets such as AR [20], implementing performance evaluation based on expression, occlusion, illumination, and their combination protocols.

- **Facial Expressions**

  Changing facial expressions leads to distortion of facial features [21], and as a consequence, a decrease in recognition quality. To evaluate the quality of recognition under conditions of facial expressions changes, the protocol E belonging to the test dataset Multi-PIE [18] is used. Images with neutral facial expressions are used as a template image for each claimed identity.

## 2.3 Low Resolution Face Recognition Methods

One of the unresolved problems [21] of face recognition is face recognition in low-resolution images. To solve this problem, 2 types of approaches have been identified: Transformation-Based and Non-transformation-based.

### 2.3.1 Super Resolution

When processing low-resolution images, one of the approaches is based on image transformation, which involves the transition to the space of high-resolution or super-resolution images [22]. However, due to the addition of new information that does not belong to the low-resolution image, additional algorithms [23, 24] are used to improve the accuracy of solving the identification problem, which optimize the transition to the super-resolution image space in order to preserve identity information.

Generative Adversarial Networks, which also provide image synthesis for training, have become one distinct trend [25]. GANs are also used to form a low-resolution training dataset, on the basis of which a feature space is identified separately for high- and low-resolution images, sharing information between networks [9]. To improve accuracy, pre-training is used, while experimenting with loss functions, for example, using triplet loss [26].

However GANs tend to produce over-smoothed images that doesn't permit to extract discriminative festures. To avoid this effect GPEN algorithm uses a method of 2 steps: learning a GAN to generate a high resolution image and embedding it into a deep neural network, and finetuning the GAN with LR face images. The results of GPEN [27] algorithm are shown at the 2. As the input the downsampled to (16, 16) and then interpolated to (112, 112) image is used. The results are compared to the original images.

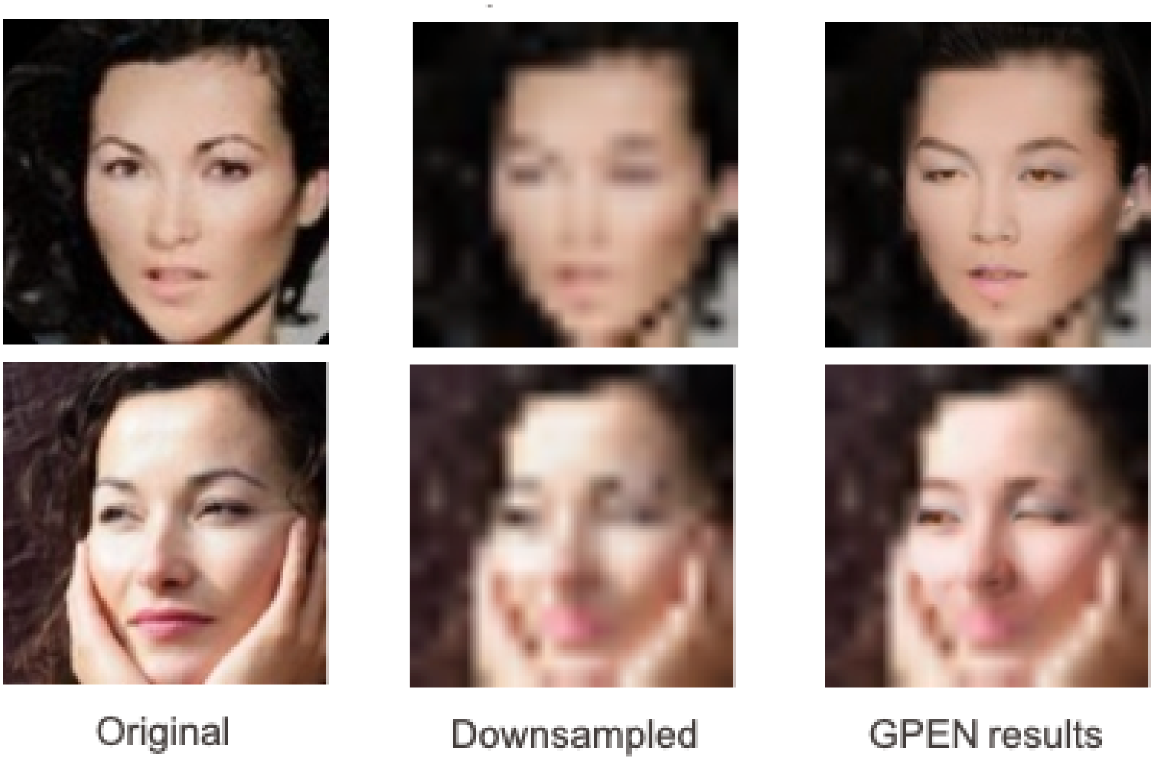

Figure 2: Results of GPEN

### 2.3.2 Non-transformation-based approaches

An alternative is to project an image of any resolution into a common feature space. For this, studies were carried out to develop the architecture of the model, the loss function, as well as the training method. With this approach, models may be trained on both high-resolution and low-resolution images, and the goal of training is to minimize the representation difference between them.

One proposed method is cross-resolution training based on a teacher-student distillation structure [28], in which the teacher processes high-resolution faces to form student distillation reference functions, and the student is trained to identify features similar to the teacher based on low-resolution data.

A separate direction is multidimensional scaling, which processes low-resolution images using feature localization and translates high- and low-resolution pairs into a single feature space [29]. To ensure the distinguishability of embeddings, discriminative multidimensional scaling is used [30].

Deep convolutional neural networks were taken as the basis for the development of model architects. Based on them, reliable partially connected networks were created [31, 32]. Gated deep networks [6] have been developed to process contextual information. Also, during their training, images of various resolutions were used [33]. In addition, to identify resolution-resistant features, it is possible to use information from different layers of the neural network by combining their contextual functions [34]. To increase the speed of inference and model learning, distributed programming methods [35], as well as a two-threaded method of convolutional neural networks with selective knowledge distillation [28], were developed.

Modern methods [36] also combine different approaches such as super-resolution, resolution matching, and multiscale template accumulation. It is also possible to adapt the loss functions, in this case the triplet loss, to obtain distinguishable representations in vector space for low-resolution images [7] and add an additional term to the main loss function, necessary to minimize the distance between high-resolution image representations and its downsampled version [37].

## 2.4 Loss Functions

From the point of view of optimization, the task of face recognition corresponds to the transformation of an image into a vector representation in a feature space, where the distance corresponds to the measure of non-similarity of faces.

- **Contrastive loss** Contrastive loss [38] involves measuring the degree of discrepancy between the obtained features for two images. If the images belong to the same class, then the discrepancy measure should be minimal, and vice versa. In addition, cosine distance can be used as a measure of discrepancy.

- **Triplet loss** To obtain distinguishable features for classification, a triplet loss [39] function was developed. It is based on the concepts of an anchor - a template image of the current personality, as well as a positive instance - an image of the same personality, and a negative instance - an image of another personality. The purpose of formulating this function is to minimize the distance between feature vectors of one class and maximize with respect to features of other classes. This leads to the need for careful formation of the pairs contained in the training dataset.

- **Center Loss** To better distinguish feature vectors, a loss function was introduced that calculates the centers for each class, Center Loss [40]. Thus, the task of optimizing this component of the loss function is to minimize the distance between the center and the embedding. This loss can be added to another loss function such as Softmax [41] to refine feature distribution.

- **Octuplet Loss** One of the methods to improve resolution stability is Octuplet Loss [7]. It is applied when fine-tuning FR models to solve the low resolution face recognition problem. This method is based on the application of Triplet Loss [39], which has shown to be effective in detecting discriminative features. By applying Triplet Loss to images and their downsampled versions, it is possible to identify the relationship between them and improve performance in cross-resolution face recognition. 3 shows the Octuplet Loss [7] algorithm.

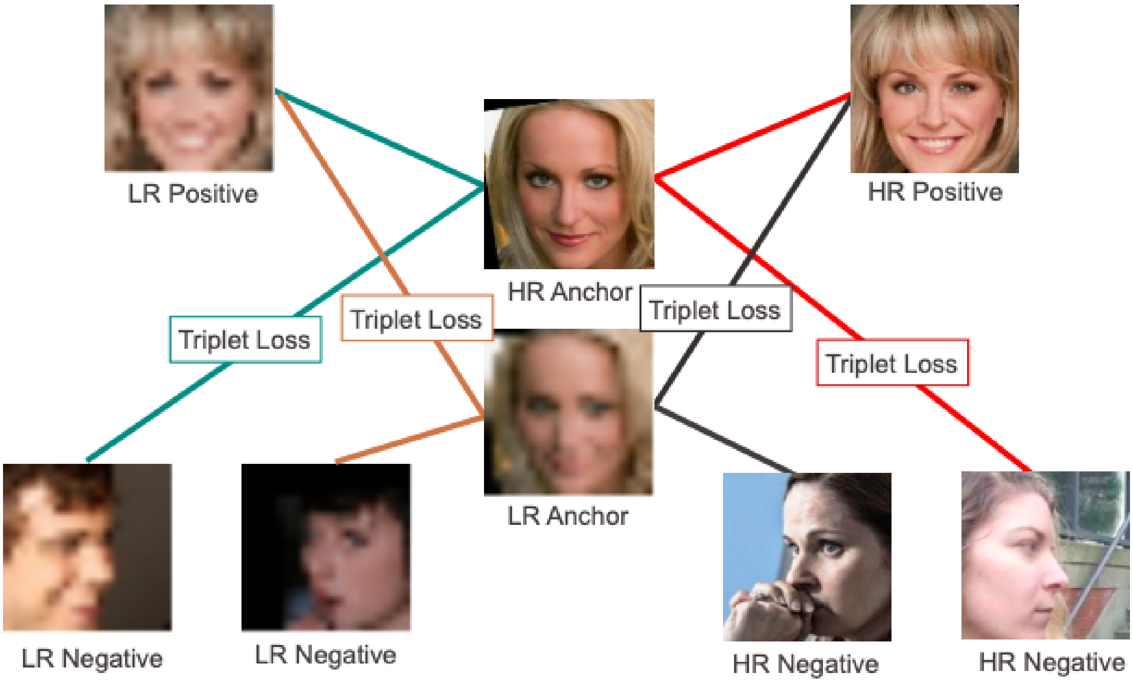

Figure 3: Octuplet Loss

- **Softmax** Since face recognition is a classification problem, the softmax [41] was also tried in the formation of embeddings. Improving the accuracy of recognition during training based on softmax [41] modifications has become one of the directions in face recognition.

### 2.4.1 Softmax-based Loss Functions

The softmax loss function can be improved by introducing a margin. The idea is that images of faces belonging to the same person should be closer to each other than images of any other classes, which in turn should be removed. Thus, the margin allows increasing intra-class compactness.

- **L-softmax** For the first time, the idea of margin was implemented in the loss function L-softmax [42] by changing the cosine of the element angles. In addition, for the first time, softmax [41] was formulated by presenting the scalar product of weights and features as the product of their norms by the cosine of the angle between them. The idea is to position the embeddings on the hypersphere in such a way as to minimize the distance between the decision boundary and the class centers.

- **A-softmax (SphereFace)** In A-softmax [43] the weight vector of each class is pre-normalized. In practice, optimization is carried out on a weighted sum of softmax and A-Softmax, the ratio of components of which changes with the number of iterations.

- **NormFace** By modifying A-softmax [43], the NormFace [44] loss function was proposed, in which, in addition to the weights, the embedding vectors were also normalized. In addition, scaling of normalized vectors has been added.

- **CosFace** By adding a margin outside the cosine,the loss function CosFace [45] was formed. Also it normalizes the weights and the feature vector and introduces the scale parameter in order to reduce radial variations. The loss function is designed in such a way that a cosine margin permits to maximize the angular margin.

- **ArcFace** In order to enhance the discriminative power of the algorithm the ArcFace [2] was proposed. It is based on the idea that embedding features are arranged around their centers on the hypersphere, so to reinforce the intra-class compactness and inter-class discrepancy there is an additive margin penalty which is equal to the geodesic distance margin penalty in the normalized sphere.

- **P2SGrad** When training models with margin-based loss functions, it is necessary to set hyperparameters that can make the learning process unstable. To solve this problem, the loss function P2SGrad [46] was designed, which formed the learning gradient adaptively.

- **CurricularFace** This loss function [47] introduces the idea of curriculum learning. In the initial stages, training is focused on simple instances. This is due to the small value of the margin. But in the course of learning, it increases in such a way that complex instances are also learned.

- **UniformFace** UniformFace [48] was also based on the idea of ArcFace [2]that vector representations of faces lie on a hypersphere manifold. The problem it solves is above large margin losses do not take into account the distribution of classes in the training sample. To solve it, it was proposed to impose a uniform distribution of class centers on the hypersphere in such a way that the distance between the broom centers of the classes is maximal.

- **MagFace** In many of the loss functions described above, the input vector and the vector of weights are normalized. However, in the studies presented in the MagFace [3] analysis, it was shown that the norm of the embedding vector correlates with image quality. It was also proved that the magnitude of the feature vector is proportional to the probability of its correct recognition. This method applies adaptive learning by modeling the distribution of classes, placing simple instances closer to the center and complex instances farther from the center.

- **AdaFace** A similar idea of correcting misclassified instances based on image quality was implemented in AdaFace [14]. In this method, an adaptive gradient change in backpropagation was introduced.

- **DeriveNet** A similar to UniformFace idea of identifying class centers was proposed by DeriveNet [49]. This method is used when fine-tuning the face recognition algorithm. The pre-trained model is used to form embeddings, which are further processed in 2 modules: reconstruction - to form a high quality image, and classification. The classification module takes as input the distance between the centers of classes, which are counted on the basis of the images reconstructed in the first module. The scheme of algorithm's work is shown at the 4.

# 3 Low Resolution Face Recognition Datasets and Evaluation Protocols

For further experiments and evaluation of the quality of recognition of face recognition algorithms, 3 datasets and their corresponding protocols were selected. In this chapter, these datasets containing images of faces in low quality, both synthesized and obtained by surveillance cameras in different shooting conditions, are considered.

## 3.1 LFW

One of the most common datasets for assessing the accuracy of face recognition is Labeled Faces in the Wild (LFW) [1]. The protocol involves matching pairs of faces. It contains over 13000 images of 1680 identities. For the LFW dataset, cross-resolution face verification accuracy is

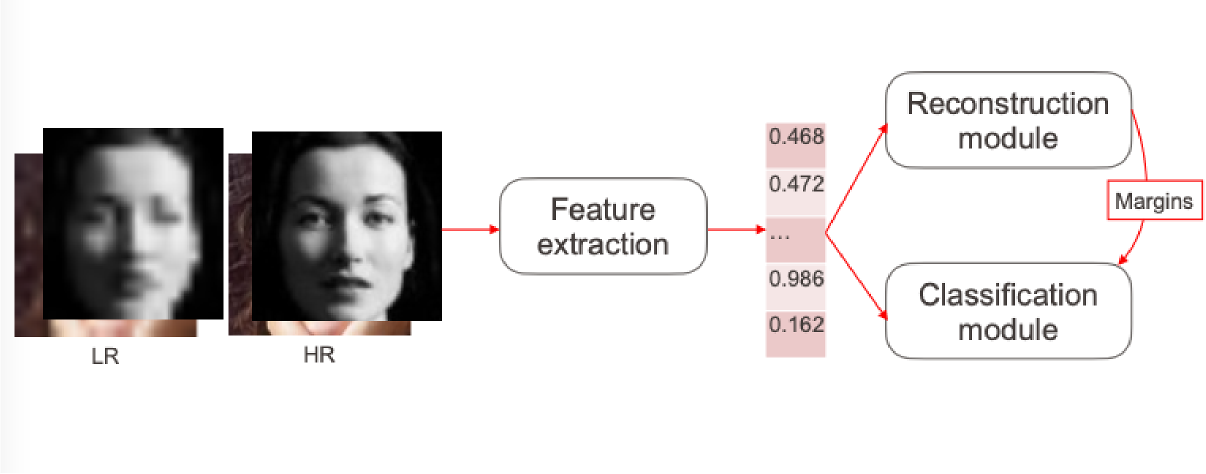

Figure 4: DeriveNet

calculated for such image sizes as (7.7), (14, 14), (28, 28), (56, 56), (112, 112). Image examples in resolutions used for cross-resolution verification are presented at the 5.

The evaluation protocol is Image Restricted Configuration [1], that is, the pairs of images processed by the algorithm are fixed in advance. The problem to be solved is the verification problem, that is, the binary classification about the belonging of 2 face images to the same person, the problems are described in more detail in the section 6.4 Evaluation.

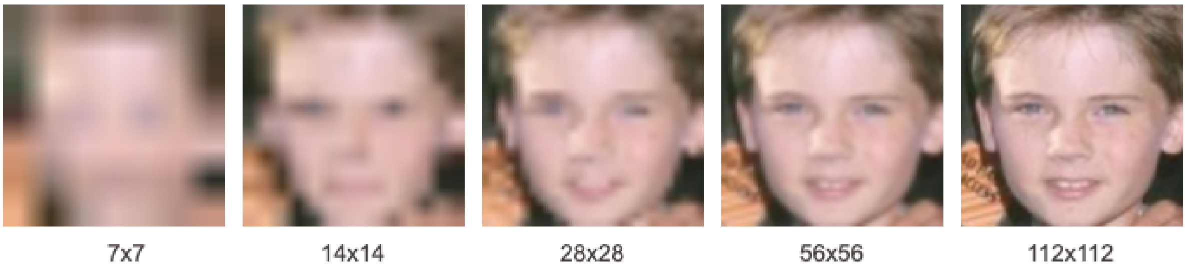

Figure 5: Example of LFW image and its downsampled variations

### 3.2 QMUL-SurvFace

The QMUL-SurvFace [13] dataset is used to evaluate the accuracy of low resolution image recognition algorithms. It includes low-resolution images obtained from real-world conditions, and not through synthesis. Images are characterized by natural obstacles, such as distractions, changes in cameras conditions, poses, and facial variations. This database contains 463,507 face images of 15,573 identities, however only a part of them is used for testing. As examples, images of different resolutions are presented at 6.

The evaluation protocol includes 2 possible configurations - solving the verification or identification problem, the detailed tasks are described in the section 6.4 Evaluation. For the experiments in this work, the verification task was chosen. As for the previous dataset, the protocol is Image Restricted Configuration [1], that is, the pair lists are fixed in advance.

### 3.3 SCface

The SCface [10] dataset includes images from 3 different surveillance cameras at 3 different distances. Each camera captures images at a low resolution, but variations in distance produce

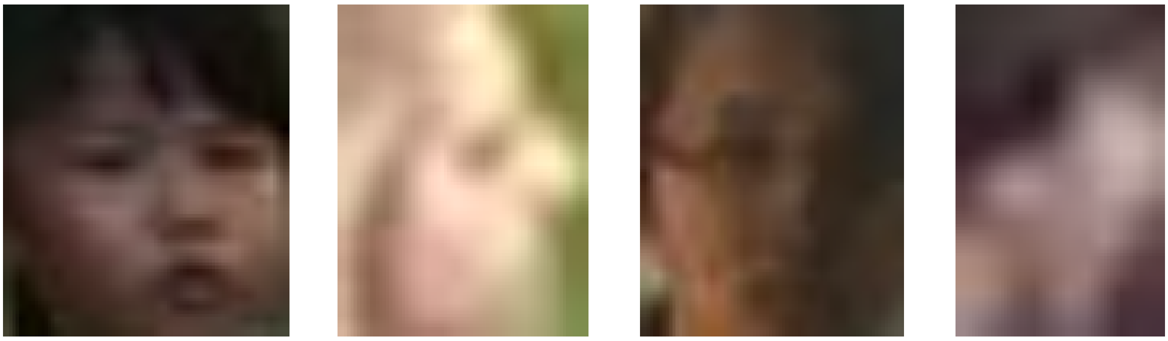

Figure 6: Example of QMUL-SurvFace images

images of varying degrees of resolution. Thus, the dataset includes 3 possibilities for assessing the quality of recognition, depending on the distance of the location of the camera on which the shooting was made. Also the dataset includes cameras for Day Time and Night Time shooting. The examples for both are presented for all 3 distances at 7 and 8. However, for the experiments only the Day Time protocol images are used.

For each protocol, there are 2 datasets dev and eval containing 44 and 43 personalities, respectively. Each identity also has a high-resolution template frontal image as a gallery image against which it will later be compared for identification. The resulting images are characterized by a high quality variability, which makes it possible to evaluate the cross resolution FR. This dataset is used in the evaluation of such low-resolution image recognition algorithms as [36].

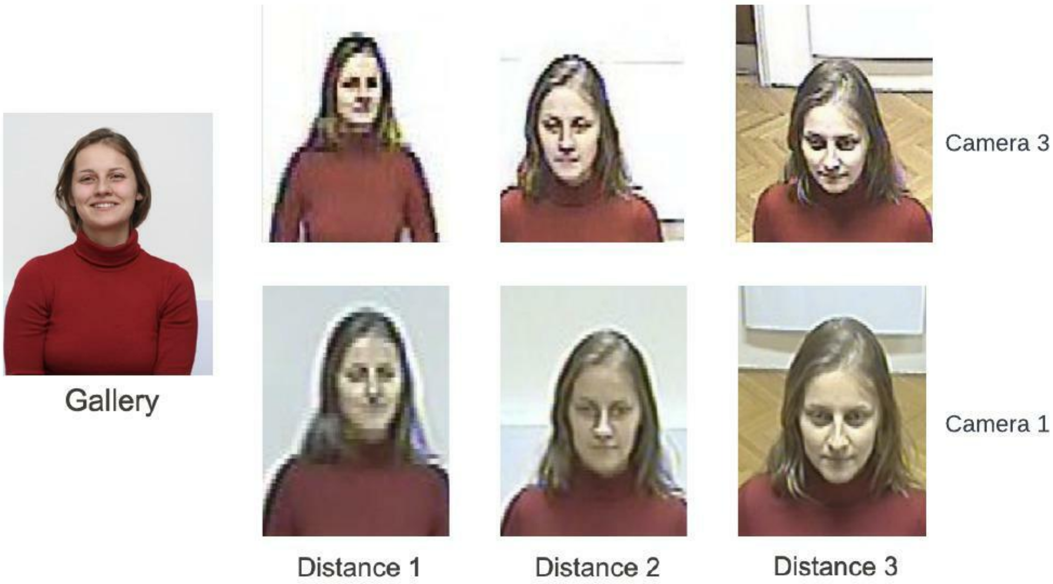

Figure 7: Examples of SCface images: DayTime protocol, distance 1, 2, 3

## 4 Implemented Low Resolution Face Recognition Methods

Some of the above methods for solving the LRFR problem were implemented as part of this work. They can be divided into 2 groups: methods that are based on high-resolution FR algorithms

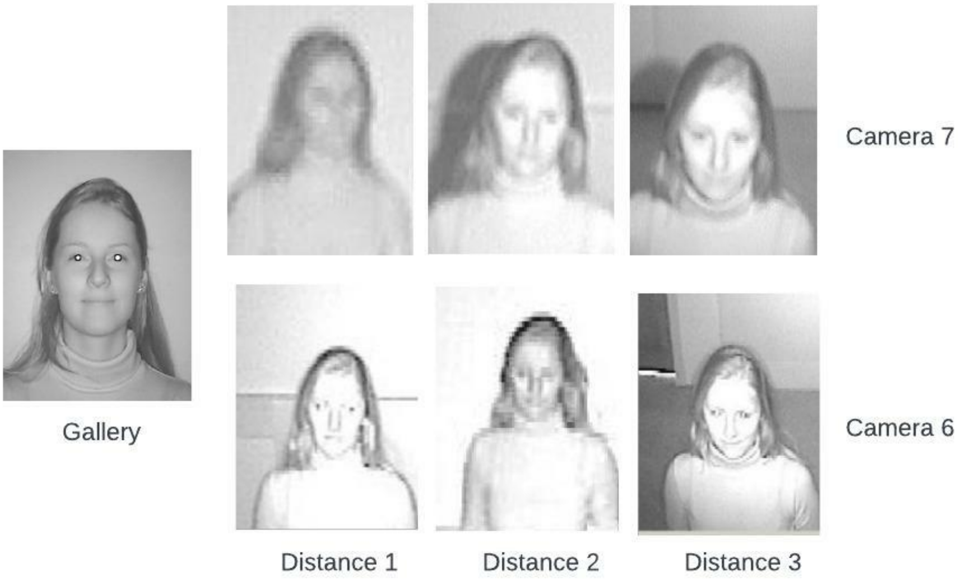

Figure 8: Examples of SCface images: NightTime protocol, distance 1, 2, 3

and fine-tuning methods.

## 4.1 Adapting High Resolution Face Recognition Methods

### 4.1.1 Cross Resolution Batch Training

In order to adapt existing methods that have shown better results in high resolution FR, there are different techniques, one of which is Cross Resolution Batch Training [4]. The idea is that each neural network processed by the dataset consists of half of the high quality images, and the other half contains downsampled images of the same dataset, but not included in the first set of high resolution images. The workflow is shown at 9. In the experiments, the downsampled images are obtained by resizing the image from (112, 112) to (16, 16) and then interpolated to the original size.

### 4.1.2 Softmax-based Loss functions

These high-resolution FR algorithms are taken: CosFace [45], SphereFace [43], ArcFace [2]. To describe their implementation, Softmax [41] can be written as follows:

$$L = -\log \frac{\exp(f(\theta_{y_i}, m))}{\exp(f(\theta_{y_i}, m)) + \sum_{j \neq y_i}^{n} \exp(s \cos \theta_j)}.$$

Then all methods differ in the internal function $f$. For each method they are presented below. This formulation, also used in AdaFace [14], will be used later to describe other algorithms.

SphereFace [43]:

$$f(\theta_j, m) = \begin{cases} s\cos(m\theta_j), & j = y_i; \\ s\cos(\theta_j), & j \neq y_i. \end{cases}$$

CosFace [45]:

$$f(\theta_j, m) = \begin{cases} s(\cos\theta_j - m), & j = y_i; \\ s\cos(\theta_j), & j \neq y_i. \end{cases}$$

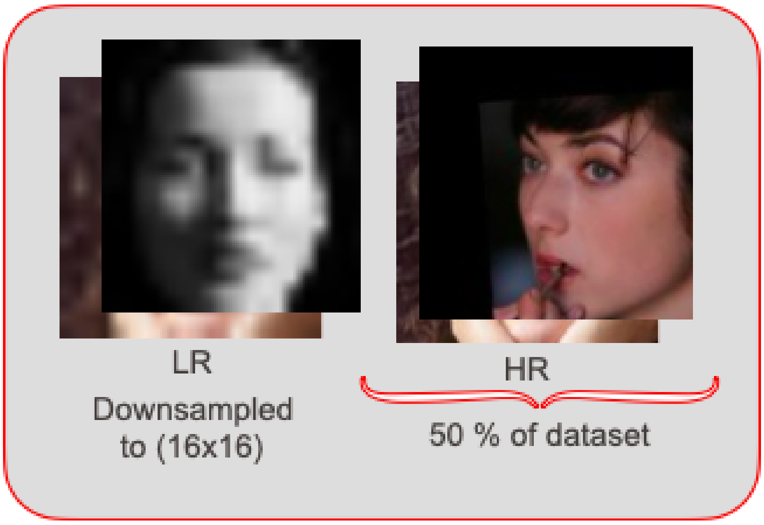

Figure 9: Cross Resolution Batch Training

ArcFace [2]:

$$f(\theta_j, m) = \begin{cases} s\cos(\theta_j + m), & j = y_i; \\ s\cos(\theta_j), & j \neq y_i. \end{cases}$$

### 4.1.3 Experiments

All experiments correspond to the training settings described in section 6.3 Training Settings. The average accuracy for cross-validation on 10 folds was taken as an estimate of the metric. The evaluation of algorithms based on the 3 datasets protocols described in section 6.4 Evaluation. Results are shown in Table 1.

| Dataset | CosFace | SphereFace | ArcFace |
|---|---|---|---|
| LFW (112x112) | 0.979 | 0.9641 | 0.9773 |
| LFW (56x56) | 0.9801 | 0.9625 | 0.97399 |
| LFW (28x28) | 0.9721 | 0.9025 | 0.96933 |
| LFW (14x14) | 0.9266 | 0.7155 | 0.9189 |
| LFW (7x7) | 0.7543 | 0.5926 | 0.74666 |
| QMUL-SurvFace | 0.6411 | 0.5953 | 0.6303 |
| SCface, dist 1 | 0.6873 | 0.1253 | 0.7349 |
| SCface, dist 2 | 0.9336 | 0.3487 | 0.95679 |
| SCface, dist 3 | 0.9229 | 0.5654 | 0.9322 |

Table 1: Comparison of the accuracy of CosFace, SphereFace and ArcFace obtained at 10 fold cross-validation

As a result, we can conclude that the best performance for LFW [1], QMUL-SurvFace [13] showed CosFace [45]. However, ArcFace [2] is slightly worse for these datasets, but shows higher accuracy for SCface [10]. The SphereFace [43] algorithm showed significantly worse results for all evaluation protocols. Thus, choosing among 3 algorithms, ArcFace [2] was the most stable in evaluation.

### 4.2 Finetuning

The 2 algorithms Octuplet Loss [7] and DeriveNet [49] were taken to solve the LRFR problem by fine-tuning the pre-trained model.

#### 4.2.1 Octuplet Loss

As described in the section 2 Related Work, Octuplet Loss [7] is based on calculating the triplet loss [39] of high-resolution images and their downsampled versions. As shown at 3, Octuplet Loss [7] includes 4 triplet loss components marked with different colors. Thus the formula is:

$$L = L_{\text{triplet}(T_{hhh})} + L_{\text{triplet}(T_{hll})} + L_{\text{triplet}(T_{lhh})} + L_{\text{triplet}(T_{lll})},$$

where $L_{\text{triplet}(T_{hhh})}$ means the triplet loss, and $T_{hhh}, T_{hll}, T_{lhh}, T_{lll}$ mean triplets formed as high resolution (HR) anchor, positive and negative images; HR anchor and low resolution (LR) positive and negative images; LR anchor and HR positive and negative images; and all images are LR, respectively. As the distance in triplet loss the cosine distance is used.

To test this algorithm, it was necessary to implement a Data Loader for generating triplets, as well as a neural network training function for getting embeddings of each triplet images.

#### 4.2.2 DeriveNet

The DeriveNet [49] algorithm consists of 2 models: Reconstruction and Classification. The scheme of the algorithm is shown at 4. A loss function is formulated for each of them: ReCent Loss for Reconstruction module and D-Margin for Classification module,

$$L_{\text{D-Margin}} = -\log \frac{\exp \|W_{y_i}\| \|x_i\| \cos \theta_{y_i} + D(C_{y_i}, C_j)}{\sum_{j=1}^{C} \exp \|W_j\| \|x_i\| \cos \theta_j + D(C_{y_i}, C_j)},$$

where $D$ is the similarity function between 2 classes;

$$L_{\text{ReCent}} = \lambda(\|\text{HR}_i - g(x_i)\|_2^2 + \|g(x_i) - C_{y_i}\|_2^2,$$

where $g$ refers to the reconstruction module and $\text{HR}_\text{i}$ is the HR image.

To test this algorithm, it was necessary to implement the modules, their loss functions and the data loader which produces the pairs of images in HR and LR.

#### 4.2.3 Experiments

Arcface [2] pretrained is used as the basis for both fine-tuning algorithms in accordance with the settings from the section 6.3 Training Settings with a training sample in high resolution. For fine-tuning, training took place over 6 epochs with settings identical to those described in the sectrion 6.3 Training Settings. The results are presented in Table 2.

As a result, when comparing 2 algorithms, DeriveNet [49] showed the best results for high resolution, and Octuplet Loss [7] for low resolution. However, since for the effectiveness of face recognition with the SCface [10] dataset protocol, it is also necessary to identify a feature vector from high-quality gallery images, for distances 2 and 3, where the test images of surveillance cameras are characterized by a higher resolution than at distance 1, the DeriveNet[49] algorithm has a higher accuracy.

## 5 Proposed Methods

Recent loss functions for face recognition imply adaptability for fitting hyperparameters [50, 51] , to prevent overfitting [47] , and to adjust the margin in the loss functions [3, 14] .

| Dataset | Octuplet Loss | DeriveNet |
|---|---|---|
| LFW (112x112) | 0.9208 | 0.9878 |
| LFW (56x56) | 0.9211 | 0.9873 |
| LFW (28x28) | 0.9135 | 0.9471 |
| LFW (14x14) | 0.8616 | 0.7103 |
| LFW (7x7) | 0.7375 | 0.5708 |
| QMUL-SurvFace | 0.6602 | 0.5618 |
| SCface, dist 1 | 0.5095 | 0.346 |
| SCface, dist 2 | 0.6296 | 0.8302 |
| SCface, dist 3 | 0.4175 | 0.9583 |

Table 2: Comparison of the accuracy of Octuplet Loss and DeriveNet obtained at 10 fold cross-validation

### 5.1 Motivation

The use of margin-based loss function modifications led to an improvement in the quality of face recognition. One of the state-of-the-art Cross Resolution Face Recognition algorithms, AdaFace [14] , proposed the idea of loss function adaptivity to image quality using margin. This method is implemented in the form of an adaptive margin function when approximating the image quality using feature norms.

In order to deal with low resolution images the training methodology of the proposed method is chosen similar to AdaFace [14]. Compared to MagFace's [3] strategy, where high norms (which are easier to recognize) are matched by high margins, AdaFace [14] can handle hard-to-find instances, such as low-resolution images. Thus, in order to improve cross resolution face recognition efficiency, a large margin needs to be matched with low quality images.

### 5.2 Approach 1

The main idea of the proposed methods is, in accordance with the articles described in the section 5.1 Motivation, the adaptation of the margin in the loss function based on Softmax [41] (in this case, ArcFace [2]) depending on the quality of the image. Since the Cross Resolution Patch Training strategy [4] described in section 4.1.1 is used for training, images are characterized by 2 resolutions: low and high, so there are 2 cases for the algorithm. For ArcFace [2] training, the optimal margin values are from 0.35 to 0.5, so these 2 extreme values will be used to adapt the ArcFace [2] to LR face recognition. Accordingly, within the framework of the softmax [41] function presented in the form from section 4.1.2, the adaptive margin function implemented in this section can be written as:

$$f(\theta_j, m) = \begin{cases} s\cos(\theta_j + m), & j = y_i; \\ s\cos(\theta_j), & j \neq y_i. \end{cases}$$

where $m = 0.35$ if the input is HR and $m = 0.5$ otherwise.

The scheme of the algorithm is shown at 10.

### 5.3 Approach 2

However in the training dataset there are images of different resolutions [52], in order to differentiate them we should use the image quality function.

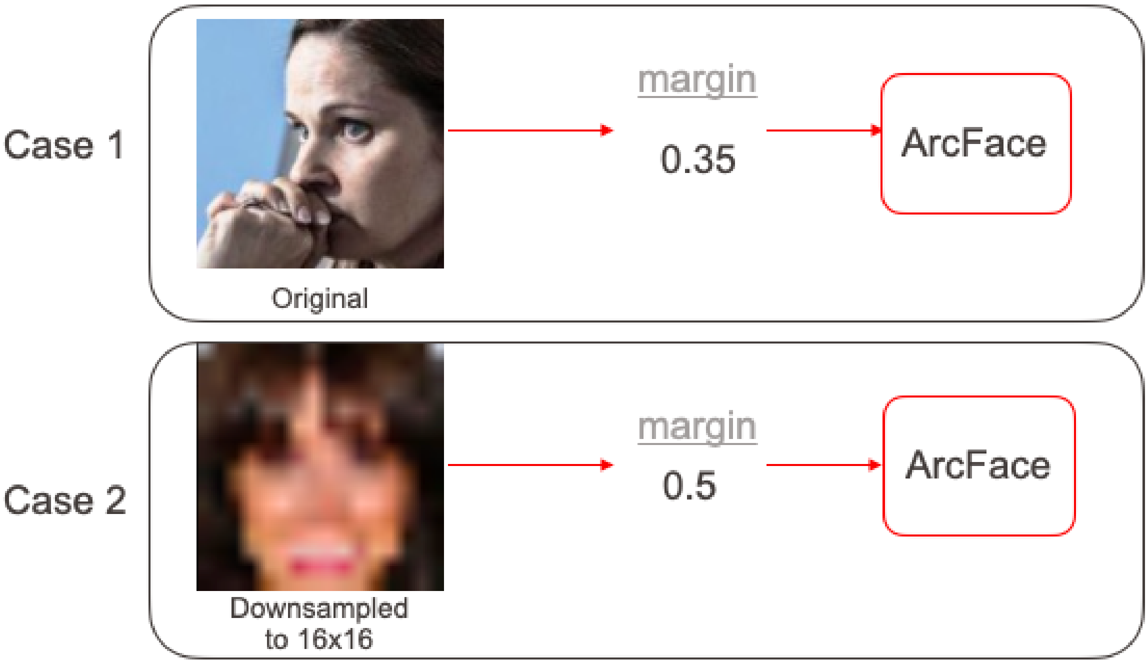

Figure 10: Proposed Method 1

### 5.3.1 Quality of Image

Image quality in this case implies the ability to be recognized. Thus, to evaluate it, it is necessary to apply methods that recognize facial features in the image, but do not require much time, as they are used in training. One of the basic methods is based on the discrete representation of the derivative. However, first order derivatives such as Sobel and Prewitt are sensitive to noise [53]. The Laplacian operator based on second derivatives does not have such a drawback [54]. To get the value expressed by 1 number, it is necessary to calculate the metric of the matrix obtained using Laplacian. In this paper, it is proposed to take the average discriminative Laplacian as the image quality metric. To demonstrate the effectiveness of this image quality evaluation function, at 11 there are examples of high and low resolution images for which norms have been calculated.

### 5.3.2 Definition

Thus, as the adaptive margin loss function, it is proposed to take the head algorithm based on ArcFace with adapted margins based on average Laplacian for processing low-resolution images. Since a high value of the Laplacian norm indicates a high image quality, it is proposed to take its reciprocal value as a margin. Thus, for low quality images, the margin will be higher. 12 shows the scheme of the algorithm.

### 5.3.3 Discussion

But for high resolution images the margin is to low, so in the future it's proposed to modify the function mapping image quality to margin in such a way that the margin value is bounded by reasonable for ArcFace [2] values. For the current approach the accuracy for high resolution is low because of non-convenient margin values.

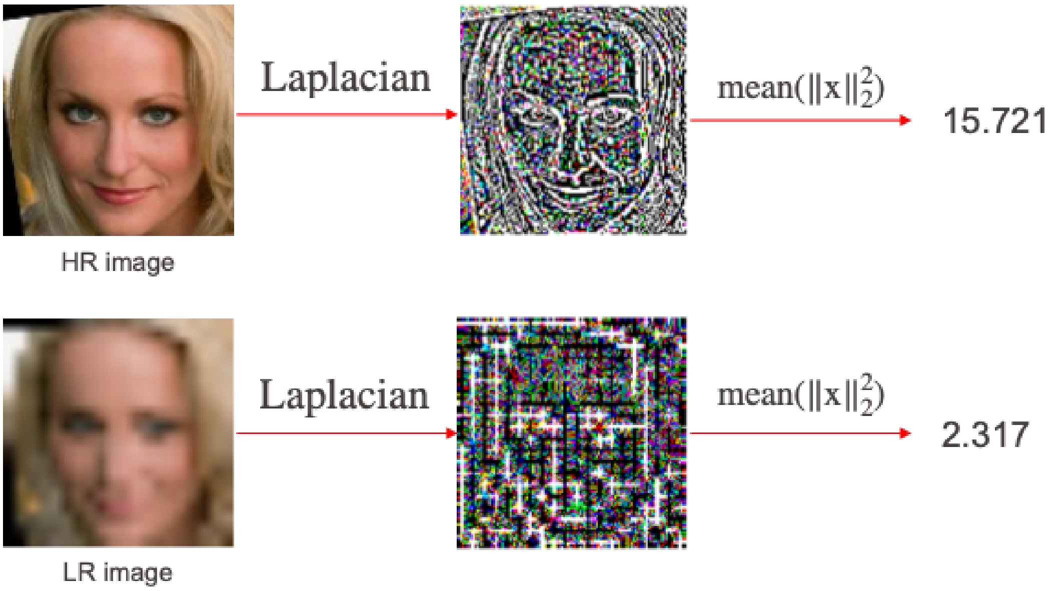

Figure 11: Comparison of Image quality using the Laplacian operator

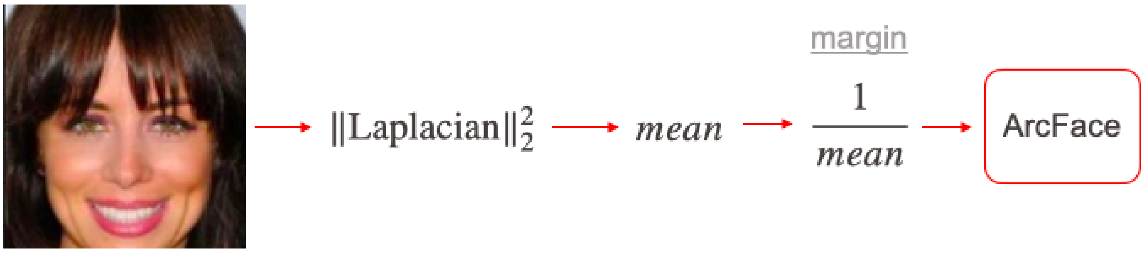

Figure 12: Proposed Method 2

# 6 Experiments

## 6.1 Training Dataset

CASIA-WebFace [52] is chosen as the training dataset. It contains 494,414 images of 10,575 real identities. The database includes images of celebrities' faces.

## 6.2 Preprocessing of training data

The MTCNN [16] algorithm is used to prepare images for both training and testing. In this way the areas of the image containing the face are obtained and aligned. Each image is resized to (112, 112) and normalized. A vertical flip with a probability of 0.5 is added as an augmentation. Half of the dataset is artificially downsampled to a resolution of (16, 16) and restored by linear interpolation to the original size in order to preserve the dimensions of the input data.

## 6.3 Training Settings

ResNet50 is used as the backbone of the neural network. Training is done using the SGD optimizer with the initial learning rate of 0.1 and learning rate step scheduling at 5, 10 and 15 epochs. The scale parameter of head neural network $s$ is equal to 64. Training lasts 18 epochs.

## 6.4 Evaluation

To test the proposed method, the verification problem is solved for 3 test datasets described in section 3 Low Resolution Face Recognition Dtasets and Evaluation Protocols.

When testing the algorithm, it is necessary to calculate the accuracy of solving the problems of verification and identification of a person from an image. As a test dataset, a set of IDs is presented, including one or more photographs of faces. After applying the trained model to the test dataset, a database of extracted embeddings is formed.

Verification is a 1-to-1 matching process that checks if the face embedding belongs to the ID class. This process is done by calculating the similarity metric between the current embedding and the ID class embedding from the embedding database. If the metric value exceeds a fixed threshold, then the image presented as the current embedding belongs to the ID class.

Identification is a task of matching 1 to N. The input is an image and a set of IDs, the output of the algorithm is the class to which the image belongs or a message that the class is not recognized. Similar to verification, the similarity metric is considered for N presented classes, among which the maximum is selected. If the maximum similarity value exceeds the threshold, then the corresponding ID is the solution to the problem.

## 6.5 Analysis

The comparison is made based on mean accuracy of 10-fold cross validation. The AdaFace [14] algorithm is chosen as the state-of-the art algorithm to compare the performance with respect to modern low-resolution face recognition methods.

### 6.5.1 Comparison with the baseline

This section presents a comparison of the proposed methods with baseline algorithm AdaFace [14] in the Table 3.

| Dataset | AdaFace | Approach1 | Approach2 |
|---|---|---|---|
| LFW (112x112) | 0.9906 | 0.977 | 0.9623 |
| LFW (56x56) | 0.9869 | 0.974 | 0.9645 |
| LFW (28x28) | 0.9685 | 0.969 | 0.9584 |
| LFW (14x14) | 0.6321 | 0.9236 | 0.9138 |
| LFW (7x7) | 0.5653 | 0.7685 | 0.7905 |
| QMUL-SurvFace, mean accuracy | 0.5324 | 0.6281 | 0.7125 |
| QMUL-SurvFace, std | 0.04875 | 0.02956 | 0.0265 |
| SCface, dist 1 | 0.50634 | 0.7539 | 0.7492 |
| SCface, dist 2 | 0.96913 | 0.9506 | 0.9182 |
| SCface, dist 3 | 0.99691 | 0.9491 | 0.8104 |

Table 3: Comparison of the accuracy of AdaFace and 2 proposed methods obtained at 10 fold cross-validation

Based on the results obtained when testing the AdaFace [14] baseline and the proposed methods, we can conclude that the proposed algorithms are applicable to solving the LR face recognition problem, since they outperformed AdaFace [14] in these tests when evaluating low-resolution images. However, in high resolution and when evaluating SCface [10], where it is necessary to process high-resolution images from the gallery, AdaFace [14] showed the best accuracy.

### 6.5.2 Comparison of all implemented methods

For clarity, the 4 is also presented containing the accuracy values for the test protocols of all the methods implemented in the work.

| Dataset | CosFace | SphereFace | ArcFace | Octuplet Loss | DeriveNet | AdaFace | Approach1 | Approach2 |
|---|---|---|---|---|---|---|---|---|
| LFW (112x112) | 0.979 | 0.9641 | 0.9773 | 0.9208 | 0.9878 | 0.9906 | 0.977 | 0.9623 |
| LFW (56x56) | 0.9801 | 0.9625 | 0.97399 | 0.9211 | 0.9873 | 0.9869 | 0.974 | 0.9645 |
| LFW (28x28) | 0.9721 | 0.9025 | 0.96933 | 0.9135 | 0.9471 | 0.9685 | 0.969 | 0.9584 |
| LFW (14x14) | 0.9266 | 0.7155 | 0.9189 | 0.8616 | 0.7103 | 0.6321 | 0.9236 | 0.9138 |
| LFW (7x7) | 0.7543 | 0.5926 | 0.74666 | 0.7375 | 0.5708 | 0.5653 | 0.7685 | 0.7905 |
| QMUL-SurvFace | 0.6411 | 0.5953 | 0.6303 | 0.6602 | 0.5618 | 0.5324 | 0.6281 | 0.7125 |
| SCface, dist 1 | 0.6873 | 0.1253 | 0.7349 | 0.5095 | 0.346 | 0.50634 | 0.7539 | 0.7492 |
| SCface, dist 2 | 0.9336 | 0.3487 | 0.95679 | 0.6296 | 0.8302 | 0.96913 | 0.9506 | 0.9182 |
| SCface, dist 3 | 0.9229 | 0.5654 | 0.9322 | 0.4175 | 0.9583 | 0.99691 | 0.9491 | 0.8104 |

Table 4: Comparison of the accuracy of all implemented methods and AdaFace obtained at 10 fold cross-validation

# 7 Conclusion

In this work, the problem of low-resolution facial recognition and methods of its solution based on deep training were studied. For the implementation of some of these methods, a face recognition pipeline for deep learning based on the existing solution was implemented. In addition, low-resolution datasets were studied: LFW, QMUL-SurvFace, SCface, for each dataset the evaluation protocol was implemented. In the course of the project, high-resolution facial recognition methods were implemented: CosFace, SphereFace, ArcFace, and adapted to solve the low-resolution facial recognition problem using the Cross Resolution Batch Training algorithm. Also algorithms for fine-tuning facial recognition model were implemented: OctupletLos, DeriveNet. Based on the studied articles and the trend towards margin adaptation in software-based loss functions, two methods were proposed, one of which sets two fixed margin values depending on whether the image is of high or low quality being processed, and the second evaluates the image quality with the help of Laplacian operator. All implemented methods were tested based on low resolution image datasets. To assess the quality of the proposed methods, the AdaFace algorithm was chosen as a baseline, which is state-of-the-art in Cross Resolution Face Recognition. As a result of the evaluation, algorithms showed accuracy higher than those of AdaFace in low resolution images.

# References

[1] 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," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[2] J. Deng, J. Guo, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," *CoRR*, vol. abs/1801.07698, 2018. [Online]. Available: http://arxiv.org/abs/1801.07698

[3] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," 2021. [Online]. Available: https://arxiv.org/abs/2103.06627

[4] M. Knoche, S. Hörmann, and G. Rigoll, "Susceptibility to image resolution in face recognition and training strategies to enhance robustness," *Leibniz Transactions on Embedded Systems*, p. Vol. 8 No. 1 (2022): Special Issue on Embedded Systems for Computer Vision, 2022. [Online]. Available: https://ojs.dagstuhl.de/index.php/lites/article/view/lites-v008-i001-a001

[5] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition," 2016. [Online]. Available: https://arxiv.org/abs/1607.08221

[6] G.-J. Qi, "Hierarchically gated deep networks for semantic segmentation," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2267–2275.

[7] M. Knoche, M. Elkadeem, S. Hörmann, and G. Rigoll, "Octuplet loss: Make face recognition robust to image resolution," 2022. [Online]. Available: https://arxiv.org/abs/2207.06726

[8] C.-Y. Low and A. Beng-Jin Teoh, "An implicit identity-extended data augmentation for low-resolution face representation learning," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3062–3076, 2022.

[9] H. Fang, W. Deng, Y. Zhong, and J. Hu, "Generate to adapt: Resolution adaption network for surveillance face recognition," in *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV.* Berlin, Heidelberg: Springer-Verlag, 2020, p. 741–758. [Online]. Available: https://doi.org/10.1007/978-3-030-58555-6_44

[10] M. Grgic, K. Delac, and S. Grgic, "Scface – surveillance cameras face database," *Multimedia Tools and Applications*, vol. 51, no. 3, pp. 863–879, 2011. [Online]. Available: http://dx.doi.org/10.1007/s11042-009-0417-2

[11] T. d. F. Pereira, D. Schmidli, Y. Linghu, X. Zhang, S. Marcel, and M. Günther, "Eight years of face recognition research: Reproducibility, achievements and open issues," 2022. [Online]. Available: https://arxiv.org/abs/2208.04040

[12] X. Wang, J. Peng, S. Zhang, B. Chen, Y. Wang, and Y. Guo, "A survey of face recognition," 2022. [Online]. Available: https://arxiv.org/abs/2212.13038

[13] Z. Cheng, X. Zhu, and S. Gong, "Surveillance face recognition challenge," *arXiv preprint arXiv:1804.09691*, 2018.

[14] M. Kim, A. K. Jain, and X. Liu, "Adaface: Quality adaptive margin for face recognition," 2022. [Online]. Available: https://arxiv.org/abs/2204.00964

[15] C.-Y. Wang, Y.-D. Lu, S.-T. Yang, and S.-H. Lai, "Patchnet: A simple face anti-spoofing framework via fine-grained patch recognition," 2022. [Online]. Available: https://arxiv.org/abs/2203.14325

[16] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multi-task cascaded convolutional networks," *CoRR*, vol. abs/1604.02878, 2016. [Online]. Available: http://arxiv.org/abs/1604.02878

[17] F. Taherkhani, V. Talreja, J. M. Dawson, M. C. Valenti, and N. M. Nasrabadi, "Profile to frontal face recognition in the wild using coupled conditional GAN," *CoRR*, vol. abs/2107.13742, 2021. [Online]. Available: https://arxiv.org/abs/2107.13742

[18] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," in *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*, 2008, pp. 1–8.

[19] N. Damer, J. H. Grebe, C. Chen, F. Boutros, F. Kirchbuchner, and A. Kuijper, "The effect of wearing a mask on face recognition performance: an exploratory study," 2020. [Online]. Available: https://arxiv.org/abs/2007.13521

[20] A. Martinez and R. Benavente, "The ar face database: Cvc technical report, 24," Tech. Rep., Jan. 1998.

[21] G. Guo and N. Zhang, "A survey on deep learning based face recognition," *Computer Vision and Image Understanding*, vol. 189, p. 102805, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314219301183

[22] J. Xin, N. Wang, X. Jiang, J. Li, X. Gao, and Z. Li, "Facial attribute capsules for noise face super resolution," *CoRR*, vol. abs/2002.06518, 2020. [Online]. Available: https://arxiv.org/abs/2002.06518

[23] F. Cheng, T. Lu, Y. Wang, and Y. Zhang, "Face super-resolution through dual-identity constraint," in *2021 IEEE International Conference on Multimedia and Expo (ICME)*, 2021, pp. 1–6.

[24] E. A. Cansizoglu, M. Jones, Z. Zhang, and A. Sullivan, "Verification of very low-resolution faces using an identity-preserving deep face super-resolution network," *CoRR*, vol. abs/1903.10974, 2019. [Online]. Available: http://arxiv.org/abs/1903.10974

[25] S. Ghosh, M. Vatsa, and R. Singh, "Suprear-net: Supervised resolution enhancement and recognition network," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 4, no. 2, pp. 185–196, 2022.

[26] N. Karim, U. Khalid, N. Meeker, and S. Samarasinghe, "Adversarial training for face recognition systems using contrastive adversarial learning and triplet loss fine-tuning," *CoRR*, vol. abs/2110.04459, 2021. [Online]. Available: https://arxiv.org/abs/2110.04459

[27] T. Yang, P. Ren, X. Xie, and L. Zhang, "Gan prior embedded network for blind face restoration in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[28] S. Ge, S. Zhao, C. Li, and J. Li, "Low-resolution face recognition in the wild via selective knowledge distillation," *CoRR*, vol. abs/1811.09998, 2018. [Online]. Available: http://arxiv.org/abs/1811.09998

[29] F. Yang, W. Yang, R. Gao, and Q. Liao, "Discriminative multidimensional scaling for low-resolution face recognition," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 388–392, 2018.

[30] F. Li and M. Jiang, "Low-resolution face recognition and feature selection based on multidimensional scaling joint l2,1-norm regularisation," *IET Biometrics*, vol. 8, no. 3, pp. 198–205, 2019. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-bmt.2018.5044

[31] Z. Wang, S. Chang, Y. Yang, D. Liu, and T. S. Huang, "Studying very low resolution recognition using deep networks," *CoRR*, vol. abs/1601.04153, 2016. [Online]. Available: http://arxiv.org/abs/1601.04153

[32] Z. Lu, X. Jiang, and A. Kot, "Deep coupled resnet for low-resolution face recognition," *IEEE Signal Processing Letters*, vol. 25, no. 4, pp. 526–530, 2018.

[33] O. A. Aghdam, B. Bozorgtabar, H. K. Ekenel, and J.-P. Thiran, "Exploring factors for improving low resolution face recognition," 2019. [Online]. Available: https://arxiv.org/abs/1907.10104

[34] G. Gao, Y. Yu, J. Yang, G. Qi, and M. Yang, "Hierarchical deep CNN feature set-based representation learning for robust cross-resolution face recognition," *CoRR*, vol. abs/2103.13851, 2021. [Online]. Available: https://arxiv.org/abs/2103.13851

[35] X. An, X. Zhu, Y. Xiao, L. Wu, M. Zhang, Y. Gao, B. Qin, D. Zhang, and Y. Fu, "Partial fc: Training 10 million identities on a single machine," 2020. [Online]. Available: https://arxiv.org/abs/2010.05222

[36] K. Grm, B. K. Özata, V. Štruc, and H. K. Ekenel, "Meet-in-the-middle: Multi-scale upsampling and matching for cross-resolution face recognition," 2022. [Online]. Available: https://arxiv.org/abs/2211.15225

[37] M. Knoche, S. Hörmann, and G. Rigoll, "Image resolution susceptibility of face recognition models," *CoRR*, vol. abs/2107.03769, 2021. [Online]. Available: https://arxiv.org/abs/2107.03769

[38] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," 2020. [Online]. Available: https://arxiv.org/abs/2004.11362

[39] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015.

[40] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 499–515.

[41] J. Bridle, "Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters," in *Advances in Neural Information Processing Systems*, D. Touretzky, Ed., vol. 2. Morgan-Kaufmann, 1989. [Online]. Available: https://proceedings.neurips.cc/paper/1989/file/0336dcbab05b9d5ad24f4333c7658a0e-Paper.pdf

[42] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1612.02295

[43] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," *CoRR*, vol. abs/1704.08063, 2017. [Online]. Available: http://arxiv.org/abs/1704.08063

[44] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "NormFace," in *Proceedings of the 25th ACM international conference on Multimedia*. ACM, oct 2017.

[45] H. Wang, Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," *CoRR*, vol. abs/1801.09414, 2018. [Online]. Available: http://arxiv.org/abs/1801.09414

[46] X. Zhang, R. Zhao, J. Yan, M. Gao, Y. Qiao, X. Wang, and H. Li, "P2sgrad: Refined gradients for optimizing deep face models," 2019. [Online]. Available: https://arxiv.org/abs/1905.02479

[47] Y. Huang, Y. Wang, Y. Tai, X. Liu, P. Shen, S. Li, J. Li, and F. Huang, "Curricularface: Adaptive curriculum learning loss for deep face recognition," *CoRR*, vol. abs/2004.00288, 2020. [Online]. Available: https://arxiv.org/abs/2004.00288

[48] Y. Duan, J. Lu, and J. Zhou, "Uniformface: Learning deep equidistributed representation for face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[49] M. Singh, S. Nagpal, R. Singh, and M. Vatsa, "Derivenet for (very) low resolution image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6569–6577, 2022.

[50] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," 2017. [Online]. Available: https://arxiv.org/abs/1708.02002

[51] X. Wang, S. Zhang, S. Wang, T. Fu, H. Shi, and T. Mei, "Mis-classified vector guided softmax loss for face recognition," 2019. [Online]. Available: https://arxiv.org/abs/1912.00833

[52] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," 2014. [Online]. Available: https://arxiv.org/abs/1411.7923

[53] R. Bansal, G. Raj, and T. Choudhury, "Blur image detection using laplacian operator and open-cv," in *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*, 2016, pp. 63–67.

[54] E. Argones Rúa, J. L. Alba Castro, and C. García Mateo, "Quality-based score normalization and frame selection for video-based person authentication," in *Biometrics and Identity Management*, B. Schouten, N. C. Juul, A. Drygajlo, and M. Tistarelli, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–9.