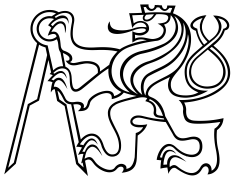
Master Thesis



Czech Technical University in Prague

F3

Faculty of Electrical Engineering
Department of Computer science

Face Image Quality Predictor

Rimma Kamaletdinova

Supervisor: Ing. Vojtech Franc, Ph.D.
Field of study: Open Informatics
Subfield: Data Science
May 2024

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Kamaletdinova Rimma** Personal ID number: **503176**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Computer Science**

Study program: **Open Informatics**

Specialisation: **Data Science**

II. Master’s thesis details

Master’s thesis title in English:

Face image quality predictor

Master’s thesis title in Czech:

Prediktor kvality obrázku tváře

Guidelines:

The objective of this project is to augment a face recognition system through the training of a neural network to assess the quality of input facial images. This face quality predictor will subsequently enhance the overall performance of the face recognition system by filtering out low-quality images during processing. The face quality predictor will be trained using existing face databases without requiring additional annotations. The evaluation of the face quality predictor will follow the protocol established by the NIST Face Image Quality challenge.
Requirements:
- Gain a comprehensive understanding of the face image quality prediction problem.
- Implement the algorithm presented in [Yermakov 2021] for training the face image quality predictor.
- Assess the trained quality predictor using the evaluation protocol outlined in the NIST Face Image Quality challenge.
- Suggest and incorporate enhancements to the implemented algorithm.

Bibliography / sources:

- A. Yermakov, V. Franc. CNN Based Predictor of Face Image Quality. ICPR Workshops and Challenges, 2021.
- Yang et al. Face Analysis Technology Evaluation (FATE) Part 11: Face Image Quality Vector Assessment. NIST Internal Report 8485.
- Grother et al. Ongoing Face Recognition Vendor Test (FRVT) Part 5: Face Image Quality Assessment. NIST Internal Report, 2022. https://pages.nist.gov/frvt/reports/quality/frvt_quality_report.pdf
- Best-Rowden, L., Jain, A.K.: Learning face image quality from human assessments. IEEE Trans. Inf. Forensics Secur. 13, 2018.

Name and workplace of master's thesis supervisor:

Ing. Vojtěch Franc, Ph.D. Machine Learning FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **23.01.2024** Deadline for master's thesis submission: _____

Assignment valid until: **21.09.2025**

Ing. Vojtěch Franc, Ph.D. Supervisor's signature	Head of department's signature	prof. Mgr. Petr Páta, Ph.D. Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

Date of assignment receipt	Student's signature

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Ing. Vojtěch Franc, Ph.D., for his significant assistance throughout my work on this thesis. I am thankful for the time he dedicated to our weekly consultations, for all his answers and advice, and for his patience. I would also like to thank my family and friends, who have always been there to support me during my studies.

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, 24. May 2024

Abstract

In this thesis, we explore the problem of face image quality prediction. The proposed approach builds upon recently proposed method for learning CNN based quality predictor (CNN-FQ) from triplets of faces [47]. This thesis enhances previous work in several key ways. Firstly, we introduce novel data selection and bootstrapping methods for efficient training on large face databases. Secondly, we train and evaluate CNN-FQ using the IARPA Janus Benchmark and CASIA-WebFace datasets. Thirdly, we explore improvements by replacing the CNN backbone with a large neural model pre-trained on the 20M LAION database. Lastly, we evaluate the quality predictors using protocols from the ongoing NIST face quality prediction challenge. Our results demonstrate the effectiveness of these enhancements, significantly improving performance over the original implementation proposed in [47].

Keywords: face image quality prediction, convolutional neural networks, bootstrapping

Supervisor: Ing. Vojtech Franc, Ph.D.

Abstrakt

V této diplomové práci zkoumáme problém predikce kvality obrazů tváře. Navrhovaný přístup vychází z nedávno navržené metody učení predikce kvality na základě CNN (CNN-FQ) z trojic tváří [47]. Tato práce rozšiřuje předchozí práci v několika klíčových ohledech. Zaprvé zavádíme nové metody výběru dat a bootstrappingu pro efektivní trénování na rozsáhlých databázích obličejů. Zadruhé trénujeme a vyhodnocujeme CNN-FQ s využitím datových sad IARPA Janus Benchmark a CASIA-WebFace. Zatřetí zkoumáme zlepšení nahrazením backbone CNN velkým neuronovým modelem předtrénovaným na databázi LAION obsahující 20 milionů obrazů. Nakonec vyhodnocujeme prediktory kvality pomocí protokolů z probíhající soutěže NIST pro predikci kvality tváře. Naše výsledky demonstrují účinnost těchto vylepšení, což výrazně zvyšuje výkon oproti původní implementaci navržené v [47].

Klíčová slova: predikce kvality obrazu tváře, konvoluční neuronové sítě, bootstrapping

Překlad názvu: Prediktor kvality obrázku tváře

Contents

Figures

Tables

Chapter 1

Introduction

The issue of human face recognition has been a significant area of interest in artificial intelligence and computer vision research for a long period. Face quality prediction, a relatively less developed field, plays a crucial role in enhancing face recognition and verification systems. High-quality images can substantially improve their accuracy and other performance metrics, whereas poor-quality images can lead to decreased metrics, non-matches, and even security problems.

The aim of this thesis is to develop an algorithm that assesses the quality of facial images to enhance the overall performance of face recognition systems. The algorithm will assign quality scores to images, filtering out those likely to result in non-matches. Building on recently published work [47], this thesis aims to enhance it further. The main contributions include training a convolutional neural network to predict face image quality using a large dataset created from a combination of the IARPA Janus Benchmark C (IJB-C) and CASIA-WebFace datasets. It is important to note that the method learns the quality predictor from existing large databases without requiring additional annotations. This thesis will present effective methods for selecting training data from a large pool, improved data preprocessing methods, and a bootstrapping method for more effective training. Additionally, an alternative approach will be implemented to extract information from images using feature vectors containing the underlying latent features of faces. Finally, the framework will be evaluated using the methodology and metrics defined by the National Institute of Standards and Technology in the Face Analysis Technology Evaluation (FATE) quality assessment track.

1.1 Face image quality

Defining image quality is essential yet complex. It can be approached in at least three ways: through industry standards, human-centric definitions, and quality concepts used in machine learning algorithms. While these methods share commonalities, each introduces unique details. Thus, image quality is inherently subjective and tied to specific tasks, research fields, or applications. In this thesis, we are mainly interested in the machine-learning definition of the problem.

Industry standards. Broadly, face image quality is described in ISO/IEC 29794-1 [19] as the degree to which a biometric sample fulfills specified requirements for a targeted application. The requirements may address aspects such as focus, resolution or the probability of achieving a correct comparison result. Other authors further specify that a sample's quality is related to being suitable for personal recognition [2] or automated matching [14]. In that case, automated matching means the system's ability to recognize/verify a person using his reference face image. The industry standard definition aims to define quality in such a way that it is associated with a high probability of a biometric system working effectively on data, such as being able to verify or recognize a person.

We already see, that the term quality contains several important aspects that can change the definition and use case of quality in some algorithm, therefore it is essential to elaborate on its components. ISO/IEC 29794-1 [19] outlines three components of quality.

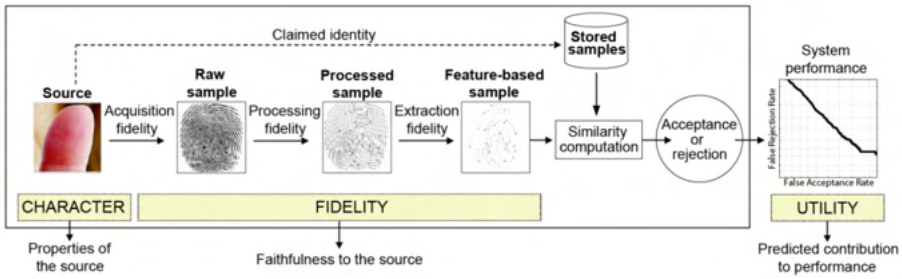


Figure 1.1: Image quality components: character, fidelity, utility. Source: [2].

As shown in Figure 1.1, the first component is character, which refers to the attributes of the individual's physical features [39]. It indicates how good the physical attributes are in terms of quality. The scars on the face, ageing that makes skin wrinkled, skin condition in general affect the character component. In cases where a face image has low character, recapturing the image will not improve its quality.

The next aspect is fidelity, which describes the similarity between a biometric sample and its source. According to ISO/IEC 29794-1 [19], the fidelity of a biometric sample includes components attributable to the environment, subject behavior, and technology. An example of a low-fidelity image could be one that is highly compressed, so that the facial features cannot be seen clearly.

The last component is utility, which is defined as the sample's impact on the biometric system's performance [39]. This is the component that is addressed by face image quality studies and algorithms in most of the cases.

Human centric definitions. From a human perspective, facial image quality is typically defined through subjective evaluation by a human observer. This involves personal judgment of various image attributes such as sharpness,

contrast, lighting, aesthetics, and the absence of blurring. It is important to note that this evaluation is highly dependent on the observer.

Machine learning quality concepts. The goal of machine learning methods is to determine a face image quality score that accurately predicts the success of a face recognition system processing those images. Approaching this problem with supervised learning is challenging due to the difficulty in obtaining data with ground truth targets, such as the probability of success. For instance, while it is easy to evaluate the success of a face matcher classifying pairs of face images, determining how individual images contribute to success or failure is much harder, in fact, it is the key challenge in this problem.

1.2 Factors affecting face image quality

The quality of face images can be influenced by different factors, which can broadly be categorized into aspects related to the person in the image, environmental factors, and technical or device characteristics [9]. Understanding and addressing these factors is crucial for improving the performance and reliability of face recognition systems, especially in critical applications like security and identification. A more detailed description of factors is presented in this section.

1.2.1 Individual-Related Factors

Individual-related factors are those connected to the person being captured or to the user of the biometric capture system. Some of these, such as facial expression or accessories, could potentially be mitigated by asking the subject to cooperate and retake a picture. However, this is not always possible, especially when using forensic or surveillance cameras [2], or in cases of aging or injuries that irreversibly change the person's appearance. We present an overview of factors in Table 1.1.

Table 1.1: Individual-Related Factors Affecting Face Image Quality

Factor	Description
Facial Expression	Variations in expression can affect facial geometry and appearance
Occlusions [22]	Objects like glasses, hats, or masks can obscure key facial features
Pose and Angle	Extreme head poses or angles can distort facial features
Ageing and injuries	Changes in facial appearance over time can affect the consistency of face recognition

1.2.2 Environmental Factors

The environmental factors in most cases address the lighting or illumination, both natural and artificial, inside buildings. Artificial lighting can be improved with appropriate equipment, but it is impossible for a person to influence natural light or weather conditions. Background complexity is another aspect mentioned: in a controlled environment, the background can be made uniform, as is the case with ID pictures. On the other hand, in photos or recordings from cameras 'in the wild,' the background is often not uniform and can indeed be problematic, especially in crowded and busy places. The factors are described in Table 1.2.

Table 1.2: Environmental Factors Affecting Face Image Quality

Factor	Description
Lighting Conditions [49]	Insufficient or excessive lighting can create shadows, reducing image clarity
Background Complexity	Busy or cluttered backgrounds can distract from the facial features and affect detection
Weather Conditions	For outdoor captures, weather elements like rain, fog, or bright sunlight can degrade image quality

1.2.3 Technical Factors

There is a wide variety of devices that can be used for capturing images and video sequences. Logically, each of these can lead to different outcomes and possible problems or distortions. For example, there is clear evidence that the use of hand-portable capture devices leads to performance degradation in face verification systems [2]. To address these issues, it is possible to provide clear instructions or methodologies for photographers, set standards for cameras and other accessory devices, in other words, standardize the capturing process. The table below (Table 1.3) illustrates the technical factors.

Table 1.3: Technical Factors Affecting Face Image Quality

Factor	Description
Camera Quality	The resolution, sensor quality, lens of the camera affect the captured data
Compression Artifacts [2]	Over-compression of digital images can lead to loss of detail and introduce artifacts
Motion Blur	Movement by the camera during capture can blur the image, reducing clarity
Focus [4]	Incorrect focus settings can result in an out-of-focus image, distorting the facial details
Exposure [43]	Overexposure or underexposure can lead to loss of detail in facial features
Noise [24]	High levels of noise, particularly in low-light conditions, can obscure facial features

1.2.4 Biometric System Factors

ISO/IEC 2382-37:2017 [20] defines a biometric system as a system designed for the biometric recognition of individuals based on their behavioral and biological characteristics. The system may include both biometric and non-biometric components, encompassing all related processes from image capture to face recognition and verification. Key factors are outlined in Table 1.4.

Table 1.4: Biometric Factors Affecting Face Image Quality

Factor	Description
Inter-System Variability	Each biometric system is unique, with a distinct combination of components, leading to variations in the qualities and other attributes it can predict
Algorithmic Sensitivity	The performance of the underlying algorithms (face detection/recognition, face image quality assessment) can widely vary and is affected by numerous factors

1.3 Use cases of face image quality

When discussing the measurement or prediction of face image quality, it is beneficial to consider the potential use cases to gain a deeper understanding of the problem. This section is divided into two parts: initially, we focus on

more abstract, general use cases of face image quality and then provide some real-world examples.

1.3.1 General use-cases

This subsection summarizes the most important use-cases of face image quality as described in [39].

1. Acquiring photos during enrollment process and database maintenance: This involves selecting the best image from a set of images taken during the enrollment process and initiating a retake if necessary to keep low-quality photos out of the database. When updating a person's template and adding a new photo, it is essential to ensure that the photo is of sufficient quality for the same reasons.

2. Summarization: Quality control across different locations or over time, for example, is possible by collecting and analyzing pictures taken at Airport A and Airport B to identify problematic locations, or in summer and winter to determine specific seasonal trends and their effect on image quality. Summarization allows for monitoring of workers/professionals, specific groups of people.

3. Additional processing: For already acquired samples of poor quality, it is possible to invoke further preprocessing, such as applying a restoration algorithm, adjusting brightness, applying a more powerful matching algorithm, or adjusting the threshold in the decision phase.

1.3.2 Real-world applications

In addition to the broader areas of application previously discussed, the following practical applications, which we encounter on a daily basis, are also important:

1. Border control systems: Automated border control systems rely on high-quality face images. The automated gates at airports are designed to be intuitive, and the lighting is uniform and symmetrical [23] to ensure the images taken are of sufficient quality and do not produce false matches, which is important for security reasons.

2. Smartphone security: Modern smartphones use facial recognition for authentication. Sometimes, when a person wears a mask or sunglasses (occlusion - worse quality), the smartphone may not authenticate immediately. Nowadays, the algorithms are quite sophisticated, however, and are trained to recognize people even when they are wearing accessories. Face authentication in smartphones is used for banking transactions and other applications as well.

3. Immigration and visa processing: There are specified standards for facial images used in visa processing. When applying for a visa and taking

a photo, sometimes the system does not accept the photo because the quality is not good enough or does not comply with the standards, allowing the person's image to be retaken immediately.

1.4 Face image quality assessment versus Image quality assessment

It is important to differentiate between Face Image Quality Assessment (FIQA) and Image Quality Assessment (IQA). As described in [34], FIQA is a specialized branch within the broader field of IQA, which focuses on evaluating the quality of images from an image processing perspective. FIQA is developed specifically for biometric applications (particularly facial recognition systems), and it primarily concentrates on the facial features that are crucial for such systems.

General IQA algorithms are less effective when applied to FIQA tasks because these general algorithms are typically designed to assess the subjective perceptual quality of images as perceived by humans. On the other hand, FIQA algorithms are designed to assess the biometric utility of face images, determining how effectively an image can be used for facial biometrics rather than how visually pleasing it is.

It is possible in some specific cases to adapt IQA algorithms for FIQA purposes, but the reverse is not true: FIQA algorithms do not generally perform well on non-facial images.

1.5 Contributions of the thesis

This thesis builds on a method for learning a face image quality predictor from triplets of faces, recently proposed by [47]. The authors demonstrated the potential of this approach using limited data in proof-of-concept experiments. The goal of this thesis is to develop an efficient implementation of this method and conduct a thorough empirical evaluation using the evaluation protocols of the ongoing NIST challenge [13]. The thesis contributions are as follows:

1. Developed an efficient implementation of the algorithm to learn the face image quality predictor, exploring different strategies for generating training triplets, which is a main challenge of the method.

2. Combined the two face image datasets, IJB-C and CASIA-WebFace, to create one large dataset, which was used for training and evaluation in one of the experiments.

3. Investigated the use of image representations from deep neural models pre-trained on large datasets, specifically using FaRL trained on a 20M database of image-text pairs [50].

4. Conducted a thorough evaluation of the developed face quality predictor using the metrics defined by the ongoing NIST challenge on face image quality assessment [13].

1.6 Outline of the thesis

Chapter 2 reviews relevant literature on face image quality assessment and existing methodologies. Chapter 3 describes the method for learning the face image quality predictor on which we build in this thesis. Chapter 4 describes enhancements proposed in this thesis, specifically triplet generation, bootstrapping, and learning using feature-vector data. Chapter 5 focuses on implementation details such as face detection, feature extraction, and the neural network architectures used by the proposed method. Chapter 6 presents the experiments and results, including dataset details, evaluation metrics, and all the experiments conducted. Chapter 7 concludes the thesis, summarizing findings and discussing implications and future research directions.

Chapter 2

Related work

This chapter reviews the existing face image quality algorithms and methods. As the NIST Face Recognition Vendor Test (FRVT) Face Image Quality Assessment has been ongoing for an extended period of time, numerous studies related to the face image quality assessment problem have been published. The official NIST Face Image Quality Assessment report [13] provides a detailed overview of face image quality and related concepts, including the metrics, algorithms, and evaluation of the chosen state-of-the-art algorithms. The most comprehensive comparison study on face image quality was published in the survey paper [34].

In general, there are several main approaches to face image quality assessment. For instance, as noted by [31], the L2-norm of an image representation extracted by a neural network trained to perform face recognition task, serves as a good predictor of face image quality; if the norm is high, the image is often visually of good quality, whereas if it is close to zero, it is probably a non-face. Other approaches focus on defining specific facial characteristics, such as exact face position, different types of occlusion, illumination, and others, as discussed in [11], [37], and [43]. Furthermore, there is a substantial body of research employing machine learning methods, both supervised and unsupervised, encompassing classical machine learning and deep learning techniques [1], [45], [21], [18], [35], [40], [30]. Some of these methods work with individual images to predict quality scores, while others involve comparing an image with a reference image to determine quality. Each of these methodologies encompasses a distinct definition of quality and its corresponding assessment criteria. We are going to focus on some of the recent studies.

2.1 A Deep Insight into Measuring Face Image Utility with General and Face-specific Image Quality Metrics

Fu et al. [10] in their recent study examine different existing methods of accesing face image quality. Specifically, they analyse the differences and results of 6 face image quality assessment (FIQA) and 10 image quality assessment (IQA) algorithms.

As for the FIQA deep learning methods, they focused on RankIQ [6] described in 2.4.3, FaceQnet [15], SER-FIQ [40], Probabilistic Face Embeddings [38], MagFace [26] and SDD-FIQA [30]. The IQA methods are divided into statistical-model-based, CNN-based, multi-task learning-based, and rank-based approaches.

To evaluate and compare all the mentioned methods, they use three databases: BioSecure [29], LFW [17] and VGGFace2 [5] and metrics such as error vs. reject characteristic (ERC), false non-match rate (FNMR) and hand-crafted metrics such as inter-eye-distance, blur, contrast, symmetry-intersection and others.

The evaluation showed that the FNMR on the BioSecure Database (which is distinct from others due to the controlled capture environment) for all the presented algorithms was very low, indicating that when all images are of high quality, almost every algorithm performs nearly perfectly. What is important to take away from this is that the data we use to build, train, and evaluate our image quality assessment system is extremely important.

Handcrafted features (metrics) were effective on the VGGFace2 database but inconsistent across different databases. These features often fail under uncontrolled conditions, such as those in the LFW database, due to their inability to adapt to specifics in image processing and different capture devices. At the same time, the behavior of metrics such as ERC allowed authors to conclude that there is indeed a correlation between the metrics and image utility, meaning the metrics were effective with respect to selecting face images with high utility. From that, we could note that it might make more sense to use some generalized metrics to assess the performance of the system.

The authors also conclude that the best option is to combine different metrics, which could lead to a more generalized measure across different FR systems and application scenarios.

2.2 Learning Face Image Quality From Human Assessments

Rowden et al. [3] proposed two methods for learning face image quality based on generated target quality scores.

To generate the scores, the authors introduced two methods: a matcher-independent method based on human assessments (HQV) of quality, and a matcher-dependent method that utilizes quality values derived from similarity scores between face images (MQV). During the human-based assessments, participants were instructed to compare pairs of face images and decide which one has better quality. As it is impossible to compare all possible pairs, each person provided responses for around a thousand pairs. Then the authors applied a matrix completion method, similar to those used in recommendation systems such as Netflix, to obtain a full set of quality scores for each participant on the entire database. For MQV, three different face matchers were used to derive the face quality from similarity scores produced

by these matchers.

The authors then used feature vectors extracted with ConvNet [42] to train the support vector regression model with a radial basis kernel function for automatic quality prediction (both MQV and HQV).

The authors evaluated the results on the IARPA Janus Benchmark-A (IJB-A) and Labeled Faces in the Wild (LFW) [17] datasets. The evaluation showed that the human assessments correlate strongly with recognition performance. Specifically, employing HQV led to significant reductions in the False Non-Match Rate (FNMR), around 2% for LFW and 13% for IJB-A for two different matchers.

2.3 Face image quality assessment based on learning to rank

Chen et al. [6] proposed a learning-to-rank approach for face quality assessment based on the framework of Parikh and Grauman [32].

The authors provide an example of testing a face recognition method on databases A and B. They state that if the performance of the algorithm on images from dataset A is overall better than on images from B, then the images from A have better quality than those in B. Moreover, it is considered that the images from the same database have similar quality values.

Next, the three image databases are introduced: one with high-quality (ID card) face images gathered in controlled environments, ”wild” or real-world image datasets LFW [17], and finally the non-face images on which the algorithm produced false-positive results. All the images were normalized to ensure only relevant facial information was used for quality assessment. These images were then used to train a model that ranks the face quality by learning from comparison outcomes across the mentioned databases.

The evaluation comprised measuring the Rank-based Quality Score (”rank weights”) on features extracted from five different feature extractors (Histogram of Oriented Gradients [7], Gabor [44], Gist [28], Local binary patterns [27], and CNN). The results showed that the images from the dataset with non-face entries were generally ranked lower, the faces without occlusion (mostly from the ”ID-card” database) were ranked higher. The method was also able to rank facial expressions, common for the second, wild, dataset, meaning it could learn and correspondingly rank the face images with different intensities of emotions; for example, the neutral facial expression had a higher RQS than the images with people showing any other kind of emotions. Additionally, the RQS scores on images with variations in pose, exposure, and resolution were explored, which allowed stating that the method is able to correctly differentiate (therefore to correctly rank) between different types and degrees of impact of these settings.

2.4 Face image quality assessment for face selection in surveillance video using convolutional neural networks

Vignesh et al. [41] address the problem of selecting high-quality images from a pool of images to enhance face recognition system (FRS) performance by imitating the FRS response using a convolutional neural network architecture.

The authors provide their own quality definition, relating it to the face recognition algorithm. They define the FR pipeline as consisting of two modules, where the first one contains face detection and localization, and the second includes face feature extraction and matching. They suggest considering the second module as a black box and model its output using a CNN. They state that the CNN should be able to predict the quality of the input images. The CNN outputs corresponding to the quality scores are used to choose the images with the highest quality.

To assess the performance of the system, they extract feature vectors from the highest-quality input images using HoG [7] and LBP [27] feature extractors. This method allows them to measure how well the CNN can predict the score of face matchers.

Unlike previous works, this one solely works with video data, specifically the ChokePoint dataset [45]. The inputs are represented as image sets. The authors calculate the feature vectors for each input sequence and the probe sequence and use the Mutual Subspace Method [46] proposed by Yamaguchi et al. to compare them. They suggest stating the image sets are similar if the canonical angle between them is smaller than the defined threshold. Next, to find the output similarity score, they compute the mean canonical angle of all the elements of the probe sequence. The resulting score helps determine whether two sets are considered a match or not.

The evaluation focuses on comparing the proposed method with other known selection approaches. The authors claim that their method is capable of selecting the best possible subset of face images and that it outperforms three of the four other methods used for comparison.

Chapter 3

Proposed method

In this study, we adopt the methodology proposed in [47]. The method trains a CNN to output a quality score for an input face image. Instead of using face images annotated with specific quality scores, which are hard to define, it uses face triplets annotated with binary labels indicating whether a given face verification algorithm correctly ranks the triplets. These binary annotations are easy to obtain. The quality scores are treated as binary latent variables in a statistical model, which is trained to predict the triplet labels. The EM algorithm estimates the model parameters, with the E-step estimating the unknown quality scores and the M-step training the CNN for quality prediction. In this thesis, we extend and enhance this method, as will be described in the following chapters. Section 3.1 focuses on the triplet ranking error. Section 3.2 elaborates on the Expectation-Maximization (EM) algorithm used for parameter learning in the model. Finally, the architecture of the system is described in Section 3.3.

3.1 Triplets ranking error

In this section, a statistical model from [47] corresponding to the described system is presented.

Consider a triplet of faces $(A, B, C) \in I^3$, where I denotes the space of input images. The triplet (A, B, C) is generated such that the images A and B capture the same individual, while C represents a different individual. Let $d\colon I \times I \to \Re$ denote the distance score computed by a pre-trained face matcher described in Section 5.2.

A face triplet (A, B, C) is assigned a binary label

$$y = [\![d(A, B) < \min(d(A, C), d(B, C))]\!] . \tag{3.1}$$

The value of the label y indicates whether the triple is correctly ranked. That is, label y equals 1 if the images are ranked correctly, meaning that the distance between the images of the same individual is less than the distance between the images of different individuals. If the ranking is incorrect, y is set to 0.

The face triplets and their associated labels are considered random variables. The distribution of the label y conditioned on the triplet (A, B, C) is described

by the following model:

$$p_\theta(y|A,B,C) = \sum_{\substack{(a,b,c)\\ \in\{0,1\}^3}} p_\theta(y \mid a,b,c)\, p_\theta(a|A)\, p_\theta(b|B)\, p_\theta(c|C) \tag{3.2}$$

where a, b, c serve as hidden variables that denote the quality of each respective image within the triplet. Finally, the probability that the face triplet will be ranked by the verification system correctly is defined as $p(y = 1 \mid A, B, C)$.

To characterize the distribution $p(x \mid X)$, the authors use the logistic distribution, which is defined based on the features extracted from the image X using the CNN-FQ.

$$p_\theta(x = 1|X) = \frac{1}{1 + \exp(-\langle \phi(X), u \rangle)} ,$$

$$p_\theta(x = 0|X) = 1 - p_\theta(x = 1|X) ,$$

with u representing the network's weights and $\phi(X)$ being the output from the final and the second-to-last layers of the CNN-FQ.

3.2 Expectation-Maximization algorithm

The foundational paper on the Expectation-Maximization (EM) algorithm, which also proves its convergence, was first presented in [33].

Let θ be a vector which encapsulates all parameters of the distribution $p(y \mid a, b, c)$ and $p(x \mid X)$.

The training set is defined as $T = \{(A_i, B_i, C_i, y_i) \subset I^3 \times \{0, 1\} \mid i = 1, \ldots, n\}$. It includes n triplets of facial images along with their corresponding labels as per equation. The goal is to optimize the model parameters θ through the maximization of the conditional log-likelihood for T, which is formulated as $L(\theta) = \sum_{i=1}^{n} \log p(y_i|A_i, B_i, C_i)$.

The Expectation-Maximization (EM) algorithm is used to transform the maximization of $L(\theta)$ into into a sequence of simpler problems. It introduces an auxiliary function

$$F(\theta, q) = \sum_{i=1}^{n} \sum_{\substack{(a,b,c)\\ \in\{0,1\}^3}} q_i(a,b,c) \log \frac{p_\theta(y_i|A_i, B_i, C_i)}{q_i(a,b,c)} \tag{3.3}$$

where $q_i(a, b, c)$ are auxiliary variables for each image triplet. The auxliary function $F(\theta, q)$ acts as a lower bound to $L(\theta)$, enabling a sequential optimization approach.

The EM algorithm alternates two steps: the E-step and the M-step. During the EM algorithm's E-step, $\max_q F(\theta^t, q)$ is calculated using a closed-form solution:

$$q_i^t(a,b,c) = \frac{p_{\theta^t}(y_i|a,b,c)\, p_{\theta^t}(a|A_i)\, p_{\theta^t}(b|B_i)\, p_{\theta^t}(c|C_i)}{\sum\limits_{\substack{(a,b,c)\\ \in\{0,1\}^3}} p_{\theta^t}(y_i|a,b,c) p_{\theta^t}(a|A_i) p_{\theta^t}(b|B_i) p_{\theta^t}(c|C_i)}$$

The subsequent M-step is decomposed into two optimization problems. The first one focuses on optimizing parameters that define the distribution $p(y|a,b,c)$. Specifically, the new iterate of the distribution is computed by

$$p_\theta^t(y|a,b,c) = \frac{1}{\sum_{i=1}^n q_i^{t-1}(a,b,c)} \sum_{i=1}^n [\![y_i = y]\!] q_i^{t-1}(a,b,c)\,.$$

while the second deals with maximizing with respect to the weights of the CNN-FQ that define the distribution $p(x|X)$. The weights are trained by maximizing the following objective function:

$$\begin{aligned} Q(\theta) = \sum_{i=1}^n \Big(& \sum_{a\in\{0,1\}} \alpha_i(a) \log p_\theta(a|A_i) \\ & + \sum_{b\in\{0,1\}} \beta_i(b) \log p_\theta(b|B_i) \\ & + \sum_{c\in\{0,1\}} \gamma_i(c) \log p_\theta(c|C_i) \Big) \end{aligned} \tag{3.4}$$

In the latter case, the optimization is directed towards a function $Q(\theta)$, analogous to training a CNN using cross-entropy loss but adjusted for soft-labels, which is resolved by the Adam optimization algorithm.

3.3 System architecture

The statistical model described in the previous section can be implemented as a neural network architecture shown in Figure 3.1.

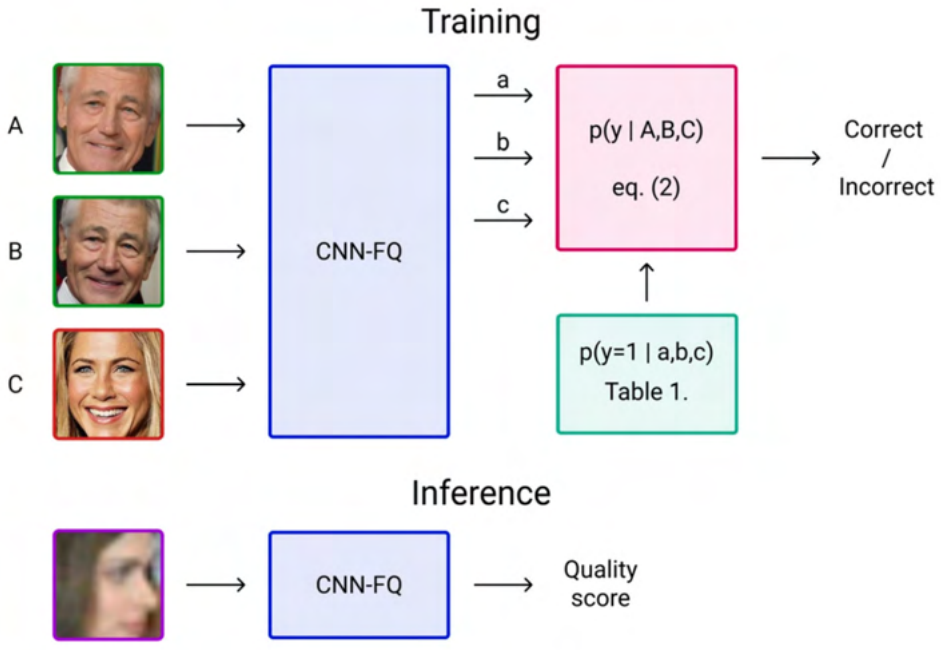


Figure 3.1: System architecture. The figure is adopted from [47].

During the training phase, triplets of images are input into the CNN-FQ. The outputs are latent quality labels (a, b, c), derived from a face triplet by the CNN-FQ. The extracted quality labels are used to compute the probability $p(y \mid A, B, C)$ as per equation 3.2, determining if the triplet is correctly ranked. The model then uses the distribution $p(y = 1 \mid a, b, c)$ to evaluate if the system correctly ranks the pairs. Figure 3.2 provides an example of such a distribution. As we can see, if all images in a triplet are of low quality, the system's ability to rank them correctly decreases. In contrast, if all images are of high quality, the system is more likely to rank them correctly.

a	b	c	$p(y = 1\|a, b, c)$
0	0	0	0.0002
0	0	1	0.0036
0	1	0	0.1534
0	1	1	0.1501
1	0	0	0.2081
1	0	1	0.1815
1	1	0	0.9999
1	1	1	0.9999

Figure 3.2: Example of distribution $p(y = 1|a, b, c)$ learned from data. Source: [47].

During the testing phase, the CNN-FQ processes an arbitrary image and outputs a quality score. This score represents the probability that the latent quality label for each image is equal to one, indicating high quality within the model's context.

Chapter 4

Proposed efficient implementation

This chapter presents the enhancements and extensions integrated into into the method for learning the face image quality predictor which was described in the previous chapter. The primary goals of these improvements are to enhance the efficiency of data generation, optimize the training process, and explore alternative approaches to improve the performance of the model.

4.1 Triplets generation

An automated binary labeling approach, used by the method described in Section 3.1 to generate inputs for the CNN-FQ, allows for the creation of an enormous number of training face image triplets. However, it is essential to generate or select data wisely to ensure that the samples represent the underlying data distribution well, therefore enabling more efficient training. The chosen data directly impact training speed, convergence, and the resulting metric values.

We implemented several strategies for triplet generation: i) random selection, ii) selection of as many unique faces as possible and iii) bootstrapping. Bootstrapping differs in that it automatically selects data during training, and is described separately in the last section of this chapter.

Random selection. In all cases we work with image templates corresponding to persons' IDs and image indices corresponding to specific faces in those images. This means that the templates may contain repeating data, while the indices do not. Random selection works in a way such that we divide the templates into train and test (validation) parts and generate the triplets using the following logic: for each unique ID (person), we find all matching entries (indices) in the dataset, compute the distances among them, sort by distance, and choose N furthest and M closest images. The non-matching image is added by randomly selecting from the pool of non-matching indices.

Unique selection. In that case, we aim to form the dataset in a way that introduces the set of already included indices and tracks it throughout the triplet-generating process, ensuring that if an index (specific face on a specific image) was already used in some triplet, we try not to append it to any other

triplet. This method was implemented with the idea of creating the most diverse dataset possible, but logically it results in a much smaller resulting dataset. We therefore allow a certain percentage of repetition in order to generate a dataset of reasonable size.

4.2 Bootstapping

The main goal of the bootstrapping implemented in this work was to allow the neural network to select the data autonomously from a large set of all face triples which can be generated from the used face database. The initial preprocessing and the main training loop remained as in the initial CNN-FQ training process. We applied several improvements to the method:

1. We used a joint CASIA+IJB-C dataset to ensure greater diversity.
2. We generated the training data using a slightly different logic: we created all possible triplets with slight bounds on the number of times a face or triplet can be used, resulting in a dataset of more than 4 million triplets.
3. The triplets dataset was randomly shuffled and divided into positive and negative triplets.
4. We created the initial dataset by choosing an equal number of positive and negative triplets.
5. Several different experiments were conducted with similar logic: train the CNN for N epochs, afterward choose a subset of new, unseen data, pass them through the CNN (in evaluation mode), identify the erroneous triplets, and add them to the initial dataset. We set a target number of positive and negative triplets for each iteration when we try to extend the dataset.
6. We added an early stopping mechanism that stops training if the validation error doesn't significantly change for 10 consecutive epochs.

The detailed explanation of experiments and their results is provided in Section 6.4.

4.3 FaRL

In addition to CNN-FQ, we decided to try another approach to predicting face image qualities. Instead of training the CNN for quality prediction from scratch, we implemented the predictor using a simple MLP on top of image representations extracted by a pre-trained neural network. Specifically, we used FaRL [50], a framework for general facial representation learning, pre-trained on the large LAION-FACE dataset containing 20 million face images, to generate the feature vectors. This model allows us to extract 512-dimensional vectors for later use.

The main difference stems from shifting from image data and CNN-FQ to the use of feature vectors extracted from these images by a pre-trained transformer and an MLP. The optimization algorithm, Expectation-Maximization, and the training loop remain the same.

Chapter 5

Implementation details

5.1 Face detection and alignment

The first step in the preprocessing pipeline is extracting bounding boxes using the RetinaFace detector [8]. When working with images from the IJB-C database, we have additional metadata available, allowing us to compute ground truth bounding boxes. For the CASIA-WebFace dataset, we simply use the bounding boxes detected by RetinaFace.

The algorithm works as follows: firstly, an input image is preprocessed. In this stage, the image can be resized based on target size. However, in our approach, we maintain the original dimensions of the image. This step is beneficial as it helps in preserving the nuances in the image data. After that, the image is normalized by subtracting a mean value from each channel, therefore standardizing the input.

Next, a RetinaFace detector is applied to the image. It identifies not only the box coordinates, but also facial landmarks, including the average positions of the mouth and both eyes. It is important to note that RetinaFace can detect multiple faces in one image. The subsequent bounding box selection step is therefore crucial.

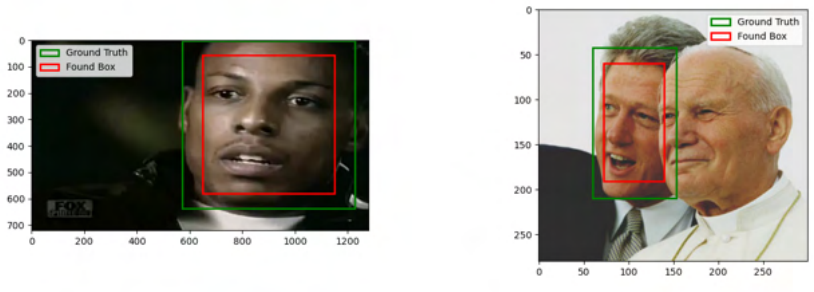


Figure 5.1: Ground truth and chosen RetinaFace boxes.

Once RetinaFace has identified all the bounding boxes, we apply an algorithm to select the most accurate one by comparing each detected box to the ground truth box, using the intersection-over-union (IoU) metric for this comparison.

This process identifies the RetinaFace box with the largest intersection regarding the ground truth box. We then compare the selected box's IoU with a predefined threshold. If the IoU is larger than the threshold, suggesting the detected box likely captures the relevant face, we add the bounding box and corresponding landmarks to our list. Examples of successfully found bounding boxes, along with the ground truth boxes, are illustrated in Figure 5.1. Otherwise, we record None for both box and landmarks for further processing.

Finally, after processing all images, we address cases where RetinaFace could not find an appropriate box and landmarks. We define an algorithm to calculate average bounding boxes and landmarks with the goal of preserving the maximal amount of information from the dataset. For bounding boxes, the algorithm computes the average offset and scaling factor based on boxes successfully extracted, enabling the generation of boxes comparable to those detected by RetinaFace. For landmarks, the algorithm calculates their average positions in relation to the detected boxes.

In the context of convolutional neural networks, it is essential to ensure consistency of input across the dataset. The subsequent alignment process involves adjusting the orientation and position of facial images to achieve this. The procedure begins with loading bounding boxes and facial landmarks detected by RetinaFace. For each image, the algorithm calculates the rotation angle needed to align the eyes horizontally, using the landmarks associated with the eye positions. The image is then rotated without cropping, which preserves the initial facial structure. Subsequently, the bounding boxes and landmarks are also rotated to match the image's new orientation.

5.2 Feature vectors extraction

To measure the distance between the identities of people depicted in two input face images, we use the extracted facial features. This distance is defined as the cosine distance between the feature vectors extracted by a neural network. As described in Section 3.1, to generate the training data, the labels are computed based on these distances.

We apply the SENet-50 model [16] to extract the vectors from each image. Again, we use the bounding boxes detected (for some images computed) by RetinaFace. The images are cropped by bounding boxes and scaled by a factor of 0.5. We resize each image to a fixed dimension (224x224 pixels) corresponding to the image used in the main CNN-FQ model and normalize the channels (again in the same way as the inputs to CNN-FQ will be normalized).

The SENet-50 model outputs a 256-dimensional feature vector for each image. Each vector is normalized to have a unit norm, which ensures that the magnitude of the feature vectors will not influence the results.

5.3 CNN-FQ

5.3.1 CNN-FQ architecture

The neural network architecture, proposed in [47] and described in Chapter 3 is defined as follows in Table 5.1.

Table 5.1: CNNFQ architecture. Each row represents a sequence of operations in the network: convolution (Conv), batch normalization (BN), ReLU activation (ReLU), sigmoid activation (Sigmoid), and average pooling (AvgPool).

Layer type	Configuration
Input	224 × 224 RGB image
Conv+BN+ReLU	64 filters, kernel: 7 × 7, stride: 2, padding: 3
MaxPool	kernel: 3 × 3, stride: 2
Conv+BN+ReLU	64 filters, kernel: 1 × 1, stride: 1
Conv+BN+ReLU	64 filters, kernel: 3 × 3, stride: 1, padding: 1
Conv+BN+ReLU	256 filters, kernel: 1 × 1, stride: 1
AdaptiveAvgPool	Global
Conv+ReLU	16 filters, kernel: 1 × 1, stride: 1
Conv+Sigmoid	256 filters, kernel: 1 × 1, stride: 1
Conv+BN+ReLU	128 filters, kernel: 1 × 1, stride: 2
Conv+BN+ReLU	128 filters, kernel: 3 × 3, stride: 1, padding: 1
Conv+BN+ReLU	512 filters, kernel: 1 × 1, stride: 1
AdaptiveAvgPool	Global
Conv+ReLU	32 filters, kernel: 1 × 1, stride: 1
Conv+Sigmoid	512 filters, kernel: 1 × 1, stride: 1
Conv+BN+ReLU	256 filters, kernel: 1 × 1, stride: 2
Conv+BN+ReLU	256 filters, kernel: 3 × 3, stride: 1, padding: 1
Conv+BN+ReLU	1024 filters, kernel: 1 × 1, stride: 1
AdaptiveAvgPool	Global
Conv+ReLU	64 filters, kernel: 1 × 1, stride: 1
Conv+Sigmoid	1024 filters, kernel: 1 × 1, stride: 1
Conv+BN+ReLU	512 filters, kernel: 1 × 1, stride: 2
Conv+BN+ReLU	512 filters, kernel: 3 × 3, stride: 1, padding: 1
Conv+BN+ReLU	2048 filters, kernel: 1 × 1, stride: 1
AdaptiveAvgPool	Global
Conv+ReLU	128 filters, kernel: 1 × 1, stride: 1
Conv+Sigmoid	2048 filters, kernel: 1 × 1, stride: 1
AvgPool	kernel: 7 × 7, stride: 1
Conv	1 filter, kernel: 1 × 1, stride: 1
Sigmoid	Output layer, output size: 1

5.3.2 CNN-FQ training

The initial stage of the training process involves preparing the input data: the images are cropped by bounding boxes extracted by the RetinaFace detector and later normalized and resized to 224x224 pixels. The random horizontal flips are applied as an augmentation technique.

The training incorporates the EM algorithm described in Section 3.2. Each EM epoch consists of three CNN epochs (a hyperparameter), corresponding to the M-step, where the network's parameters are being maximized. For optimizing the CNN-FQ's parameters, the Adam optimizer and the learning rate of 0.0001 are chosen. In the subsequent E-step, the latent variables are updated based on the newly computed values.

Throughout the training, we compute and monitor various metrics, including training and validation errors, the number of predicted false positives and false negatives, loss function and the probability tables as described in Section 3.3. These metrics provide insights into the model's performance over epochs. Additionally, we track the values of F and L described in Section 3.2.

5.4 FaRL architecture

The proposed FaRL method uses a simple multilayer perceptron architecture with three hidden layers. The architectural details are presented in Table 5.2.

Table 5.2: MLP architecture. Each row represents a sequence of operations in the network: fully connected layers (FC) followed by activation functions ReLU or Sigmoid.

Layer type	Configuration
Input	Input feature vector, input size: 512
FC + ReLU	128 neurons, input size: 512
FC + ReLU	128 neurons, input size: 128
FC + Sigmoid	1 neuron, input size: 128, output size: 1

Chapter 6

Experiments and Results

This chapter describes experiments done with the method implementing all improvements proposed in this thesis. In Section 6.1 we first describe datasets used in our experiments. In section 6.2 we present the evaluation metrics used to evaluate the algorithms. Section 6.3 provides the experiments along with their results and discussion.

6.1 Data

6.1.1 IJB-C dataset

The IARPA Janus Benchmark-C [25] (IJB-C) dataset, established in 2018, is a fundamental benchmark for evaluating facial recognition technologies. It consists of facial images and videos from 3,531 individuals, including 31,334 still images and 117,542 frames from 11,779 videos. On average, each subject is represented by 6 images and 3 videos. The dataset authors ensure that every individual appears in at least two images and one video. Figure 6.1 shows example images from the IJB-C dataset.

Figure 6.1: Sample images from IJB-C. Source: [25].

The videos in the dataset were sourced from YouTube and they mainly focus on individuals. Information about age, gender, and geographic region was collected through the Wikipedia API. Google and Wikimedia Commons were used to gather images, while videos under the Creative Commons license were

obtained from YouTube. Workers from Amazon Mechanical Turk provided annotations and bounding box data.

In contrast to predecessor datasets, IJB-A and IJB-B, IJB-C provides subjects in various poses and with different types of face occlusions, which introduces more complexity. The dataset also aims to move beyond a focus on celebrities or individuals in similar, appearance-based professions by representing a wide spectrum of the global population and including individuals from different occupations.

The dataset provides detailed metadata for each image and video frame, including details about occlusions, subject and file identifiers, facial coordinates, dimensions, environmental settings, along with demographic information and pose angles.

6.1.2 CASIA-WebFace dataset

The CASIA-WebFace dataset, created by Yi, Dong et al. [48], was developed with the idea that nowadays data are more important than the algorithms themselves and to overcome the problem that many large-scale training datasets are private. It consists of 10,575 individuals and 494,414 images. Each subject in the dataset has at least 15 images. The authors ensure that the subjects in CASIA-WebFace do not overlap with the LFW dataset [17] by checking for name duplicates. Example images from the CASIA-WebFace dataset are illustrated in Figure 6.2.

Figure 6.2: Sample images from CASIA-WebFace dataset. Generated with own script.

The images are crawled from IMDb (Internet Movie Database—an online database of information related to films, TV series, and starring celebrities)

in a semi-automatic way: firstly, the names of celebrities born from 1940 to 2014 are gathered. Subsequently, the images available in the gallery for each celebrity are crawled. To assign the correct identity, authors group the photos based on names and then apply a clustering method to label all the photos. After that, the images are cropped using found bounding boxes and saved into individual folders.

As mentioned by the authors, the incorrect annotations from CASIA-WebFace were removed manually. However, it is important to note that the dataset still contains some errors: images in the dataset are divided into individual folders with photos, and in some of them, different people may appear. We suggest ignoring this, because statistically, there should not be that many false matches.

6.2 Evaluation metrics

This section outlines the metrics used to evaluate the algorithm presented earlier. We adopt the evaluation methodology from the FRVT NIST Face Image Quality Assessment track.

The quality scores should predict the matching outcome when comparing two images of the same person. ISO/IEC 2382-37 [20] defines False Non-Match Rate as the proportion of completed biometric mated comparison trials that result in a false non-match, where biometric mated comparison trials involve comparing a biometric probe (sample input) with a biometric reference (an existing reference image, for example, a face image stored digitally on a passport). This implies the need to compute matching scores for pairs of images, then compare the predicted quality values with these scores to compute the metrics.

The terms mated, matching, similarity, and genuine scores are used interchangeably in this chapter and refer to scores computed from the pair of images of the same person as described above.

The evaluation assumes working with N image pairs, x_{i1}, x_{i2}. To compute the genuine scores, s_i, the algorithm described in Section 5.2 is used. Algorithm F introduced in Chapter 3 is used to calculate quality scores for individual images, resulting in quality vectors $q_{i1} = F(x_{i1})$ and $q_{i2} = F(x_{i2})$.

NIST applies two distinct quality assessment strategies depending on the application setup. The Application-Webcam photos assume having a frontal face image as the reference and a lower-quality webcam probe image. In that case, the vector of quality values is formed as $q_i = q_{i2}$, considering only the quality of the probe image itself. In the wild image dataset, where both images exhibit widely varying quality, the minimum of the quality values in the pair is chosen, i.e., $q_i = \min(q_{i1}, q_{i2})$. This assumes that a low comparison score is likely caused by the image with the lower quality.

Having the similarity vector s_i and quality vector q_i computed for N image pairs x_{i1}, x_{i2}, the evaluation metrics described below are computed. The formulas are fully adopted from the FRVT NIST quality assessment track.

6.2.1 False Non-Match Rate

Referring to the ISO/IEC 2382-37 [20] FNMR definition, we may further specify that the false non-match rate denotes the frequency at which a biometric matching system incorrectly categorizes two signals (images) originating from the same individual as if they were from different individuals [36].

Given N genuine scores s_i and N quality scores q_i computed beforehand, we construct the error/reject curve. The recognition threshold T is set to divide the scores into true accepts, $s_i \geq T$, and false rejects, $s_i < T$. Then, FNMR is recomputed using different fractions, r, of low quality images excluded from the computation.

The quantity is then

$$\mathrm{FNMR}(r) = \frac{\sum_{i=1}^{N} H(q_i - Q)(1 - H(s_i - T))}{\sum_{i=1}^{N} H(q_i - Q)} \tag{6.1}$$

where the numerator represents the situation of having a good-quality image with the threshold at or above Q and at the same time resulting in a false non-match corresponding to being below threshold T, and the denominator denoting the count of images with good quality, i.e., that are not discarded. To compute the quality threshold Q, the inverse of the cumulative distribution of the N quality values is taken. The similarity threshold T is computed based on predefined false-match rate. The terms FMR and FNMR are inversely related, therefore T is balancing the acceptance of true matches against the rejection of false ones. A lower FNMR score indicates better performance.

FNMR: A Practical Illustration

The goal of this section is to show that excluding low-quality images should decrease the FNMR.

We refer to Figure 6.3. Each dot in the images represents an image. Above each dot, we have the quality scores; the images are sorted by quality scores from highest to lowest. Below each dot, we have the similarity scores, which are not sorted. In simple terms, the FNMR numerator represents the situation where the quality score q for an image is greater than or equal to the quality threshold Q and the similarity score s is less than the similarity/recognition threshold T. The denominator is the count of images where the quality score q is greater than or equal to the quality threshold Q.

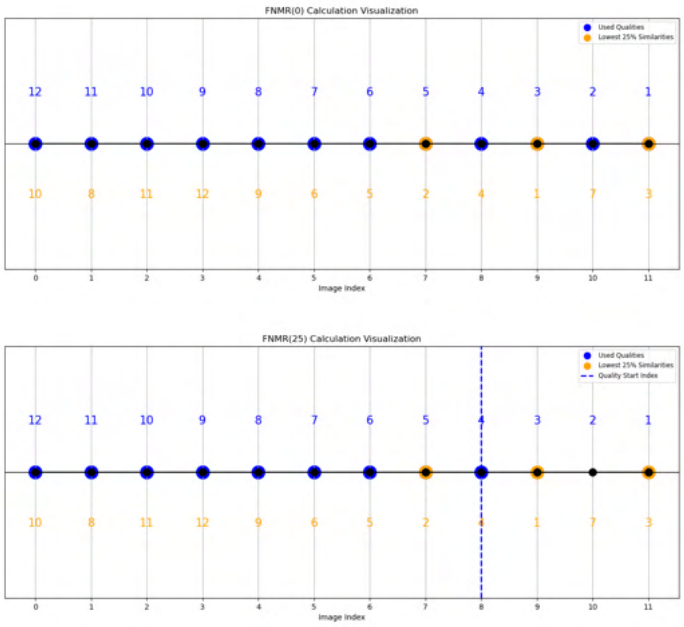


Figure 6.3: FNMR computation visualisation for different r.

In first situation, we set the quality threshold Q to 0, meaning all images will be included. We set the recognition threshold T to 25%, meaning the 25% of lowest similarity scores will be excluded. The blue points represent the used qualities, and the orange points represent the chosen similarities (those below the threshold).

- The numerator is 3 because there are three cases where a point's quality score is above the threshold Q and the similarity score is below the threshold T.
- The denominator is 12, as the number of images above the threshold Q equals the total number of images.

In second situation, we set the quality threshold Q to the 25th percentile. The blue dotted line indicates the index of the image from which $q \geq Q$ is true, meaning we only consider images above this quality threshold. The blue points represent the used qualities, and the orange dots represent the chosen similarities (those below the threshold).

- The numerator is 1 because there is only one point where the quality score is above the threshold Q and the similarity score is below the threshold T.
- The denominator is 9, as we have excluded the 25% of low-quality images (3 images).

6.2.2 Efficiency

The efficiency score is computed by normalizing the False Non-Match Rate (FNMR) with respect to the factor of excluded images, r. The optimal value is 1.

$$\eta(r) = \frac{1}{r}\left(\frac{\mathrm{FNMR}(0) - \mathrm{FNMR}(r)}{\mathrm{FNMR}(0)}\right) \tag{6.2}$$

6.2.3 Incorrect Sample Rejection Rate

The Incorrect Sample Rejection Rate (ISRR) is defined as

$$\mathrm{ISRR}(Q) = \frac{1}{N}\sum_{i=1}^{N}(1 - H(q_i - Q))H(s_i - T) \tag{6.3}$$

i.e. the proportion of samples with quality below threshold Q, and genuine similarity score at or above recognition threshold T. This error rate reflects the inaccurate rejection of a photo, where the system incorrectly assigns low quality to an image that would be correctly matched by a face recognition engine. The optimal value is the lowest.

6.2.4 Incorrect Sample Acceptance Rate

The Incorrect Sample Acceptance Rate (ISAR) is defined as

$$\mathrm{ISAR}(Q) = \frac{1}{N}\sum_{i=1}^{N}H(q_i - Q)(1 - H(s_i - T)) \tag{6.4}$$

i.e., the fraction of samples with quality above the threshold Q that at the same time have a genuine score below the recognition threshold T. This error rate reflects the incorrect assignment of high quality to an image (therefore accepting it) while having a negative outcome (similarity below the recognition threshold) in the recognition process. Again, the optimal value is the lowest.

6.3 Experiments and Results

This section presents the conducted experiments along with their configurations and results. We start with the experiment that we set as a baseline in Section 6.3.2 and subsequently present the different ideas for improving the training process or data generation. Section 6.4 elaborates on the bootstrapping method presented in Section 4.2. We try out the different configurations of bootstrapping, finally choosing the best one and running it on the merged dataset of IJB-C and CASIA-WebFace. In Section 6.5, we present the results of training of the multilayered perceptron working with vector data instead of image data, as proposed in Section 4.3. We finally compare all the methods and choose the best one in Section 6.7, along with the discussion of the results.

When evaluating, we focus on several metrics. Along with comparing the training and validation errors, the conditional log-likelihood L, the auxiliary function F, we apply the evaluation methodology from the FRVT NIST face quality challenge. In addition, we evaluate the ability of the trained models to correctly assign the quality score and to correctly order the scored images.

For full-size images with qualities predicted by the algorithms, please refer to the Appendix B.

6.3.1 Test dataset

As the FRVT NIST protocol proposes evaluating the performance of the quality-prediction algorithms on face pairs, we created the dataset as follows.

The test dataset was created in advance in such a way that we initially divided the IJB-C dataset into two parts so that they do not intersect. The images were chosen both from frames and images to ensure the representativity of the dataset, similar to the training data. We ended up with having 10,362 unique faces which corresponds to 5,182 unique pairs of face images. The test set is used exclusively for calculating NIST evaluation metrics and for assigning quality scores for visualization of sorted qualities for human assessment.

6.3.2 Baseline experiment

The goal of the baseline experiment is to evaluate the performance of the original CNN-FQ proposed in [47], providing a benchmark for the improvements we aim to achieve. In the baseline experiment, we use images with bounding boxes extracted using RetinaFace. We do not apply any additional data preprocessing or transformations here. The training and validation triplets are generated using feature vectors extracted by SENet-50, by finding M closest and N furthest pairs and randomly adding the non-match to them, as described in Section 4.1.

The IJB-C dataset is used here as a source of images with a total of 3,496 different identities (persons). We divide the dataset into train and validation using the ratio 0.2. The triplet generation results in 42,400 training triplets with around 66% of unique faces, where exactly half are positive triplets and the other half negative triplets. For validation triplets, totaling 18,216 with 56% uniqueness, positive and negative triplets are again evenly divided.

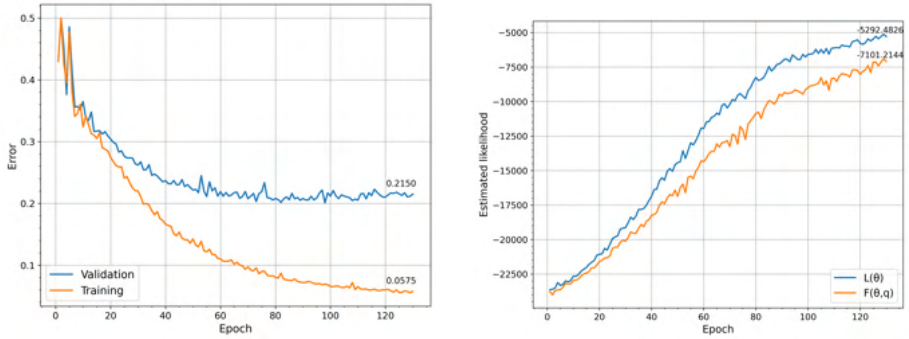


Figure 6.4: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for the Baseline experiment.

During the training process, we track the errors, L, and F. As we can see from the plots, it takes more than 100 epochs to converge, which may be logical considering the sizes of the training and validation subsets. The training and validation errors are computed for each triplet by comparing the ground truth label and the predicted label (here, a threshold of 0.5 is set to convert the predicted probability of a triplet being ranked correctly into a binary label). As presented in Figure 6.4 on the left, initially, both training and validation errors start around 0.5 - it is the moment when the model predicts the same label for every triplet. As the training proceeds, we see a monotonic decrease in both errors, resulting in 5.75% training and 21.5% validation error. The log-likelihood, L, and F are monotonically increasing, and $L \geq F$, which gives us the right to say that the algorithm works as intended.

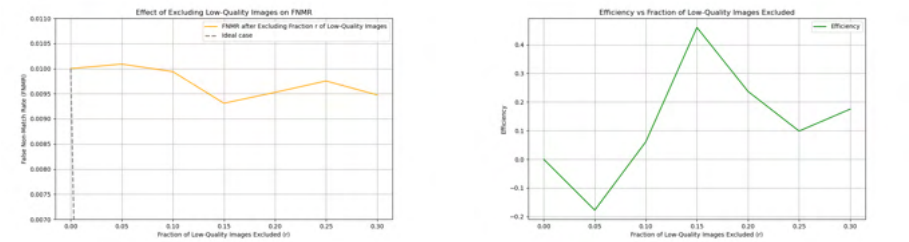


Figure 6.5: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Baseline experiment.

Figure 6.5 shows the False Non-Match Rate (FNMR) and the efficiency, which is FNMR normalized with respect to the factor of excluded images, as defined in Section 6.2.2. The initial threshold is set to give an FNMR of 0.01, i.e., the lowest 1 percent of matching scores. The scores are computed by comparing images. We show the metrics for different rates of images discarded: from 1% to 30%. In general, we assume that with low-quality images being discarded, the FNMR should decrease while the efficiency, on

the other hand, should increase. It is clear that the behavior of both metrics is fluctuative. If we compare their initial value and values at 30% of low-quality data removed, we see improvement: the FNMR decreases from 0.01 to 0.0095 and the efficiency increases from 0 to 0.17. However, in an ideal case, the metrics would be monotonic. The gray line on the FNMR plot represents the ideal case of how the FNMR should behave. Of course, we may have expected the FNMR to be closer to the ideal line, but the results of some SOTA algorithms presented in [12] show the same trend.

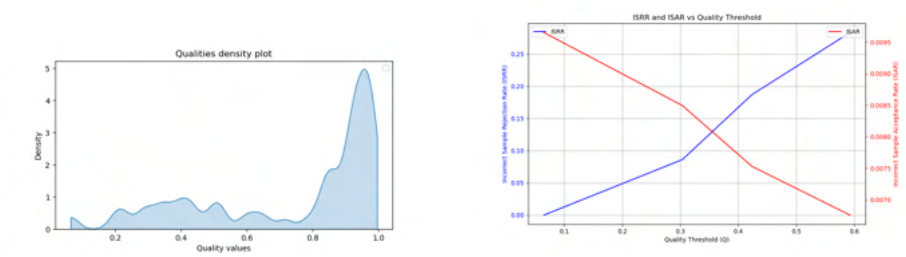


Figure 6.6: Qualities, ISRR, ISAR for Baseline experiment.

In Figure 6.6, we can see the distribution of the predicted values on the left and the plot depicting the trade-off between ISAR and ISRR on the right.

As we can see from the quality plot, the trained model tends to predict mostly higher scores for the face images. It may sound good, but we need to be critical here, as the IJB-C dataset is very challenging, with different types of occlusions. The high scores may not always match the images that are really of high quality.

Analyzing the ISAR, corresponding to the acceptance of incorrect samples, and ISRR, corresponding to the rejection of incorrect samples, allows us to find the balance between the two errors - in practice, we would likely want to achieve this result. The quality threshold here means a minimum acceptable quality for samples in the dataset. With the threshold increasing, more samples get rejected and fewer get accepted. In this particular case, the optimal point is around 0.35 with the corresponding values of ISRR and ISAR being around 0.12 and 0.008, respectively.

To better understand the ability of the algorithm to predict quality, we present Figure 6.7 with the quality values sorted from the lowest to the highest. It is clear that the lower-quality images mostly depict people with their faces turned to the side, and the pictures are either of lower resolution or blurred, while the higher-quality images mostly show people looking directly at the camera. The ordering is not perfect, so we may see some occluded or non-frontal faces among those with higher scores.

Figure 6.7: Images sorted by quality for Baseline experiment.

Figure 6.8: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the Baseline experiment.

Figure 6.8 presents several cases of individuals with images arranged in ascending order of quality. These plots suggest that the ordering by quality for individuals works. Additionally, an examination of the exact predicted quality values may be beneficial. For instance, in image (a), from a subjective human perspective, the second image appears to be of higher quality due to the facial features seen clearly; however, the assigned quality score of 0.965 may be too high for an image of such resolution. In image (b), the stability of the algorithm's predictions is evident, particularly within the same individual. Given that these images likely originate from a video file, it is expected that the quality scores would be similar, which aligns with the observations. Finally, image (c) demonstrates that while the assignment of quality to the highest quality image seems appropriate, the ordering may not be always optimal. A manual assessment by a human might place the first and second images in a different order.

6.3.3 Evaluation of impact through generation of unique triplets

As in the previous case, we make use of the bounding boxes from RetinaFace for training and feature vectors extracted by SENet-50 for triplet generation. The current experiment differs from the baseline in the way of generating triplet data: here, we try to make a dataset as diverse as possible while keeping its size reasonable as described in Section 4.1, resulting in approximately 9,000 training triplets with around 80% unique faces and validation triplets with the size of approximately 2,100 and 90% uniqueness.

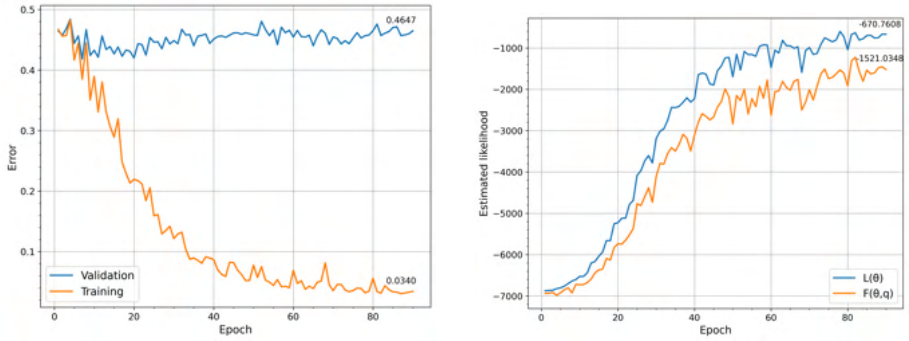


Figure 6.9: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for Unique triplet generation experiment.

From the Figure 6.9 it is obvious that the current experiment differs from the baseline. Even though we expected the uniqueness of the dataset to positively affect the evaluated metrics, the situation is the opposite. The training error decreases to 3.4%, but the validation error does not decrease during the whole training process. The key assumption behind this behavior is the insufficiently large training dataset. It is possible that the model fits well to the training set that is not representative enough, so overfitting occurs and results in relatively poor performance on the validation data. Overfitting can also arise from a situation where the model is too complex for the given data. Another assumption is that different combinations of face images may be beneficial for the model, but with the current script, it was not possible to obtain such triplets due to the fact that some of the faces were already included in another triplet. As for the metrics F and L, we observe the desired tendency to increase.

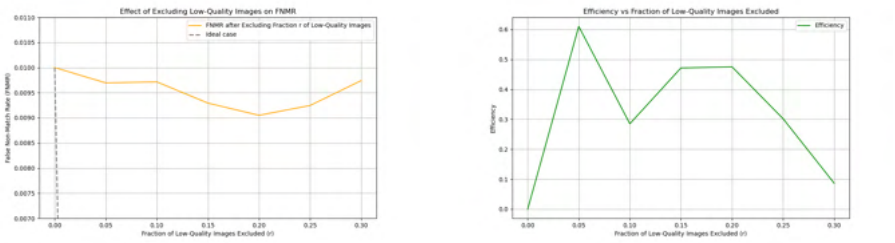


Figure 6.10: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Unique triplet generation experiment.

The experiment results for FNMR and efficiency are depicted in Figure 6.10. Analyzing both metrics together gives us a clear understanding that the model does not work as intended. Indeed, we see a decrease in FNMR and an increase in efficiency for some smaller ratios of excluded low-quality images, but after this certain ratio corresponding to 20%, both metrics start to degrade. We expect the metrics to behave monotonically - with the ratio growing, the FNMR should decrease and the efficiency should increase. This non-monotonic tendency can signal that the quality scores do not effectively represent the genuine scores.

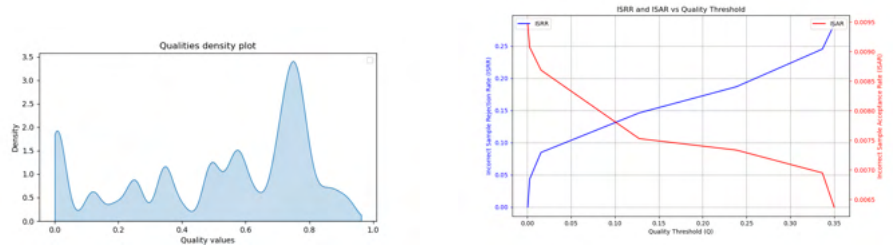


Figure 6.11: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Unique triplet generation experiment.

The distribution of quality scores presented on the left in Figure 6.11 suggests that the model is able to predict diverse scores across the dataset, which may be more optimal than predicting only the high or low scores, because the dataset contains images with a wide variety of qualities. Moreover, we expect the IJB-C to contain some reasonable amount of low-quality scores, and the current model setting allows for this. The ISAR-ISRR tradeoff on the right proposes the quality threshold of 0.1 as being the optimal one. In this case the value is quite low, which means that system is very selective and starts to reject the samples even at quality threshold 0.1. ISRR in the optimal point has a value of 0.12 and ISAR of 0.008.

Figure 6.12: Images sorted by quality for Unique triplet generation experiment.

As observed in Figure 6.12, the sorting of all images by quality does not show a clear trend as before; we can see blurred, low-resolution, or high-saturation images among those that were assigned high quality. On the other hand, there are examples of subjectively good-quality faces that were added between the poor ones. Overall, it is harder to spot any specific trend, and therefore the ordering does not appear efficient enough.

Figure 6.13: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the Unique triplet generation experiment.

When we address the examples of individual quality assignment and ordering, presented in Figure 6.13, there are several important things to notice. Image a) suggests that the model can correctly assign low quality to poor images and high quality to better ones. If we look closer at the first four pictures, we might expect the scores for them to be close to each other, as they originate from the same video sequence. However, the variance in predicted scores is significant; it ranges from 0.0157 to 0.485, even though the images appear to have similar low quality. Image b) demonstrates the impact of occlusion on the predicted quality: the first three images are partially occluded by a microphone, whereas the last one depicts a normal face picture.

On the other hand, the specific quality values are questionable, as the last image is relatively low-resolution and the person is looking to the side. The last example, plot c), presents a subjectively good quality assignment and ordering.

6.3.4 Impact of using face alignment on the CNN-FQ performance

To further improve the results of training, we proposed introducing additional data preprocessing and alignment. The first change was in bounding box detection: as described in Section 5.1, we decided not to discard the images where RetinaFace was not able to find a face. Instead, we computed the bounding box for such images and added the landmarks computation as well. The other suggestion was to align the images to standardize the inputs to the convolutional neural network. It is commonly known that alignment of inputs is beneficial and can improve the performance. Here we work with 53,294 training triplets, 62% of uniqueness; and 20,700 validation triplets, 53% of uniqueness.

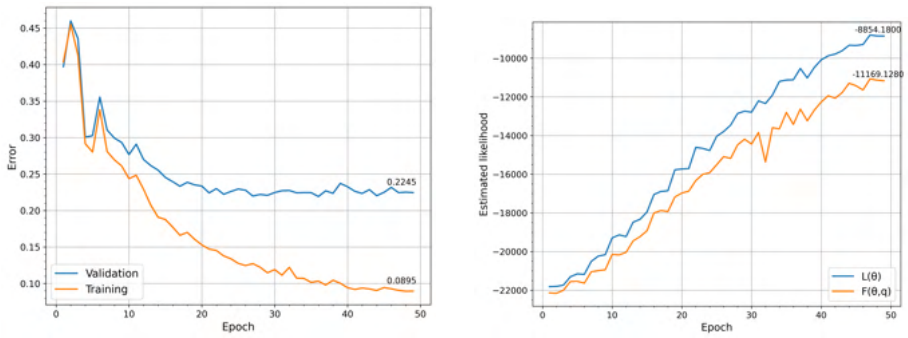


Figure 6.14: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for Aligned experiment.

The error and F,L plots are depicted in Figure 6.14. In contrast to our expectations, the results of the model with additional improvements did not lead to better performance. This outcome may be logical in terms of computing the average bounding boxes and landmarks, because statistically, there were not many cases where RetinaFace failed to detect a bounding box: only around 5% of the input data was prone to this problem. As for the alignment, unfortunately, this step did not improve the model's results. It is possible that the alignment process removed some valuable information from the data, the initial model was already performing near its optimal capability, or the initial images were already standardized enough, meaning the alignment did not change them much. The results from the experiment show a training error of 8.95% and a validation error of 22.45%. The log-likelihood L and log-likelihood lower bound F performed as expected, similar to the previous

experiments.

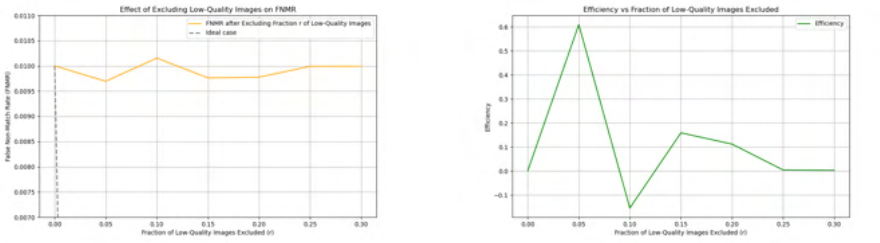


Figure 6.15: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Aligned experiment.

As in the baseline model, both metrics, FNMR and efficiency, in the current experiment (Figure 6.15) show some fluctuations. There is no clear trend of monotonic decrease in FNMR and increase in efficiency. Moreover, if we look at their values at 30% of excluded low-quality images, they return to the initial values (0.01 for FNMR and 0 for efficiency), which indicates that excluding such a high percentage of images is not beneficial. However, in NIST reports, they often consider a range of excluded images from 0% to 10% only, so we do not know if this behavior is problematic or can be normal for some algorithms.

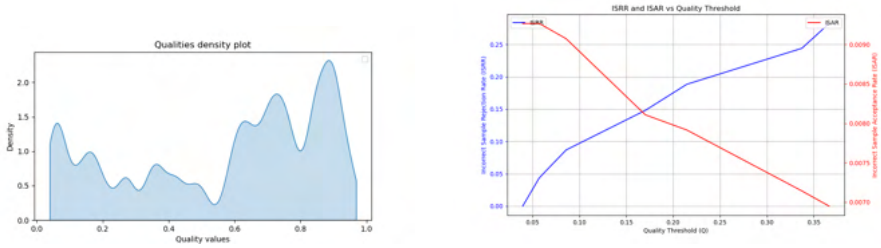


Figure 6.16: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Aligned experiment.

As seen in Figure 6.16 on the left, the quality density plot shows a significant difference in predictions compared to the baseline experiment. Instead of predicting high values in most cases, we see peaks at the lowest values between 0 and 0.2 and the highest values between 0.8 and 1. However, the distribution is non-uniform, which implies that simply excluding low-quality images might not be straightforward, as different quality clusters may impact the model differently. The optimal quality threshold, with a value around 0.15, suggests that images with a quality below 0.15 are considered low quality and excluded. ISRR has a value of 0.14 and ISAR of 0.0081.

Figure 6.17 presents sorted images for the Alignment experiment. As in the baseline model, it is clear that the images assigned a high quality represent frontal pictures of faces and images with higher resolution. At the

Figure 6.17: Images sorted by quality for Aligned experiment.

same time, we see that this is not always true: there are a few examples with visibly lower quality or non-frontal poses. Lower quality images have poor lighting conditions, blur, and noise. However, it is important to note that the ordering is not perfect, and some of the predicted quality scores do not seem reasonable.

Figure 6.18: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the Aligned experiment.

Moving on to analyzing the quality predictions for individuals in Figure 6.18:

Picture a) shows the ordering that may be found valid from a human perspective: the quality of the first three images is lower with the person facing to the side. Moreover, with the quality increasing for these pictures, more facial features are seen. The highest-quality image is clear, with relatively good quality and a close-to-frontal pose. The only irregularity is the difference in score between the first image and the subsequent two; it is several times lower, even though visually they are similar.

For the individual in b), we could say that the quality assignment is

connected to the emotions. While the first pictures depict a person during a speech, the last one shows a neutral face.

Finally, for the last person, we may deduce that the highest-quality image got this score because there are no glasses and the facial expression is again almost neutral.

6.3.5 Experiment with additional data preprocessing, alignment and unique triplets generation script

The experiment described in this section was based on data preprocessed in the same way as in Section 6.3.4, meaning the use of additionally computed bounding boxes, landmarks, and alignment of images. In addition, we used the same triplets generation script as in Section 6.3.3, which led to a training dataset of size 9,000 with 80% unique faces and a validation dataset of size 2,000 with 80% unique faces.

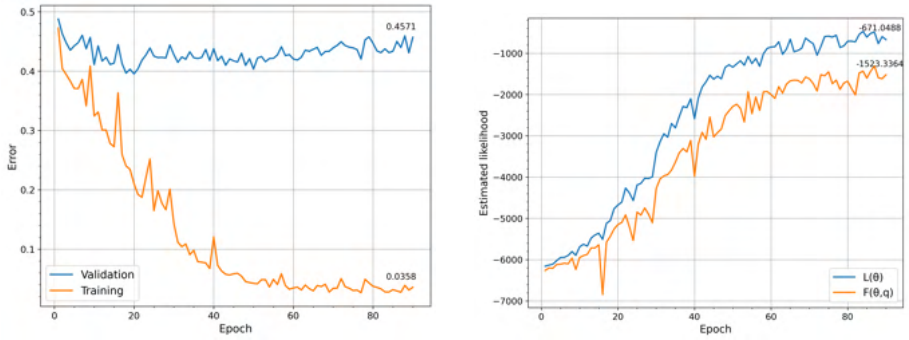


Figure 6.19: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for Aligned Unique experiment.

Given the smaller dataset size compared to a similar experiment without alignment and other preprocessing steps, the results are as expected, with clear signs of overfitting. Refer to Figure 6.19 for the details. The training error is 3.58%, and the validation error is 45.71%, closely matching the results in Section 6.3.3. This indicates that neither the additional preprocessing nor the different triplets generation logic was successful. Even though L and F show the desired growing trend, we cannot say that the experiment overall resulted in any improvement based on these metrics.

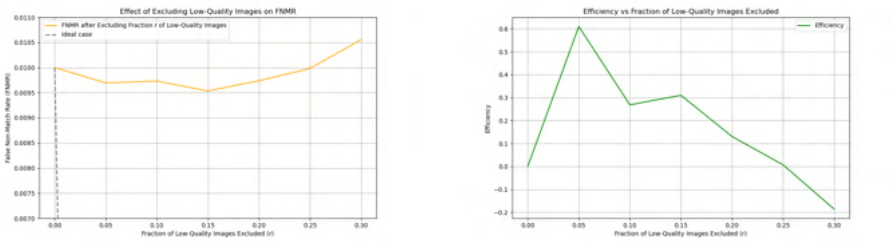


Figure 6.20: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Aligned Unique experiment.

As we proposed to evaluate the models using several approaches, the study of NIST metrics (Figure 6.20) may show different results. This is because the metrics computed during training, such as error, do not always directly reflect the overall behavior of the model, meaning the predicted qualities may still be reasonable. On the contrary, the FNMR shows an increasing trend and the efficiency starts to drastically decrease, both from the quality threshold of 0.15. The model is therefore incapable of predicting the quality scores that reflect the true qualities.

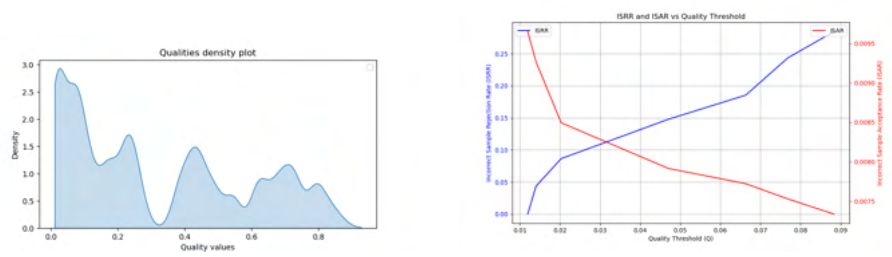


Figure 6.21: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Aligned Unique experiment.

From Figure 6.21 on the left, it is obvious that the quality density plot is shifted to the left, meaning the model mostly predicts low values, with a significant concentration around 0. At the same time, it fails to predict values over 0.8, which is a problem because the IJB-C dataset contains high-quality images among others. The right figure suggests choosing a value around 0.03 as the optimal point to balance between incorrect acceptance and incorrect rejection, which is the lowest threshold across the presented models. This may be logical given the quality distribution, with a peak around [0, 0.2]. We end up with ISRR 0.11 and ISAR 0.0082.

Given the poor performance based on all the metrics used, it is not expected that the ordering of qualities will be accurate. However, we still present the predicted qualities in Figure 6.22. We cannot see a clear or consistent trend in predicting the qualities; high- and low-quality images appear both among those scored with low and high-quality values, indicating randomness in the quality assessment process. The quality values do not visually correlate with

Figure 6.22: Images sorted by quality for Aligned Unique experiment.

parameters such as lighting, blur, or occlusion.

Figure 6.23: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the Aligned Unique experiment.

Despite the poor overall results of the model, its quality predictions presented in Figure 6.23 are reasonable for some individuals. For the person in plot a), the first image is blurred, low-resolution, and shows a non-neutral face with a non-frontal pose. The best image is also non-frontal, but the lighting and resolution are visibly better. In plot b), the first image is occluded, underexposed, and the hat occludes the face. In the second image, even with the glasses on, the quality seems better as the underexposure is no longer a problem, and the man looks directly at the camera. For the person in plot c), the first three images are blurred and have poor lighting, leading the model to rate them lower than the last image, which is high-resolution, frontal, and has no brightness issues.

6.4 Bootstrapping experiments

6.4.1 Choosing the best configuration

We decided to conduct a bootstrapping experiment on the larger dataset, with the hypothesis that increasing the dataset size should improve the results. We combined the IJB-C and CASIA-WebFace datasets. The identities in the datasets did not overlap. The preprocessing steps, such as bounding box detection, were exactly the same for all the data to ensure consistency of inputs. The resulting dataset contained around 14,000 identities and 600,000 faces (images).

Section 4.2 described the bootstrapping process and triplets selection logic. Here, we conduct several experiments to choose the best hyperparameters, namely the size (the target number of positive and negative new data to add in each evaluation iteration) and the frequency of updates. Table 7.1 presents the configuration of three experiments. The initial dataset size is 20,000 triplets for each experiment, as this seemed reasonable based on previous experiments. The ratio of positive to negative data is always 50:50, unless stated otherwise. The hypothesis is that each experiment is more complex and should lead to better results. Table 6.1 shows the configurations for the experiments.

Table 6.1: Bootstrapping experiments configuration

Initial dataset size	Target number of P/N	Frequency of updates
20,000	2,000	Every 15th epoch, from 15 to 90
20,000	5,000	Epochs 15, 30, 45, 60, 80
20,000	5,000	Epoch 3, then every 9th epoch from 9 to 90

We present in Figures 6.24 and 6.25 the results from the training only, as the main idea was to choose the best configuration and run it until convergence. Therefore, we provide only the metric plots that are tracked during training. We compare experiment 1 with experiment 2, and experiment 2 with experiment 3, for better visibility. We stopped the models' training when the trend of potential improvement in validation error was clear. Around epoch 70, experiment 1 had a validation error of 21.92%, experiment 2 had 19.8%, and experiment 3 had 18.49%. The graphs are downloaded from Weights and Biases (wandb), a tool and platform for tracking machine learning experiments. One wandb step corresponds to approximately 4 training epochs.

Figure 6.24: Bootstrapping for Experiment 1 and Experiment 2.

Figure 6.25: Bootstrapping for Experiment 2 and Experiment 3.

6.4.2 Best configuration results evaluation

The initial dataset size was 20,000 triplets, with 10,000 positives and the same number of negatives. We added an additional 5,000 positives and negatives in epoch 3 (to ensure the functionality of the code) and then in each epoch divisible by 9. It is clear that in later epochs, the model is more capable of correctly predicting the right label for a triplet. Therefore, when adding new data aiming to choose 5,000 positives or negatives, we set a 'max_tries' parameter to 10 to prevent infinite loops and to manage time complexity. In all cases, this parameter value was sufficient to add the desired amount of data. By the end, after epoch 90, where the last evaluation and addition iteration was made, we ended up with a dataset of around 130,000 triplets. Having 130,000 triplets means processing around 390,000 images in every epoch, which is highly time- and memory-consuming. Therefore, we do not train for more than 100 epochs.

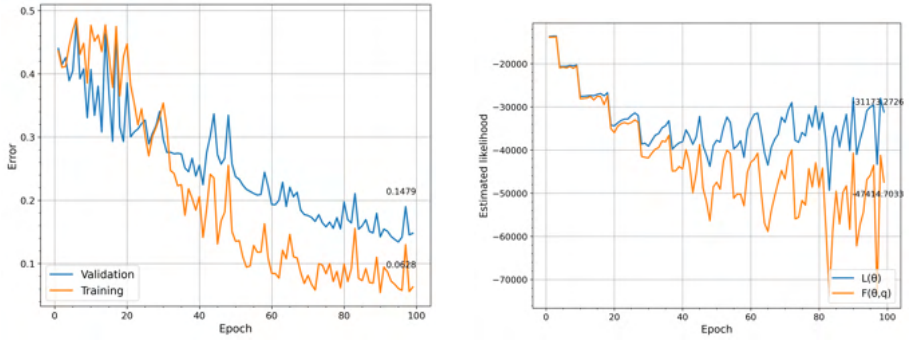


Figure 6.26: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for Bootstrapping experiment.

Figure 6.26 displays errors and F, L for Bootstrapping experiment. The fluctuative behavior of both the errors and F, L is defined by the bootstrapping strategy: when we add new data, which the model identified as erroneous, it is clear that the model will not immediately correctly predict the labels. Therefore, in each evaluation and addition iteration, the errors will increase, and L, F will decrease.

When creating the joint dataset of IJB-C and CASIA, we anticipated that the larger dataset size would significantly improve training results. As shown in the error plots, the training error decreased to 6.28% and the validation error to 14.79%. Compared to previous experiments, there is a notable improvement: the validation error decreased by 6.71%. The training errors are similar in both cases. This outcome indicates that using a larger and more diverse dataset, which includes faces from both IJB-C and CASIA, is essential for enhancing performance.

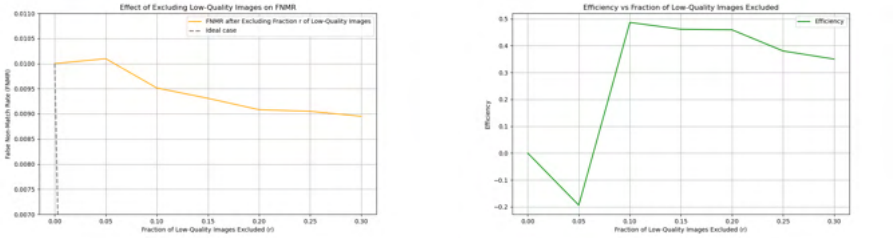


Figure 6.27: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Bootstrapping experiment.

In analyzing the FNMR and efficiency (Figure 6.27), we generally track two things: monotonicity and the correct direction of metrics. Here, the FNMR and efficiency are not strictly monotonic, but the fluctuations of FNMR are negligible. The FNMR shows a visible decreasing trend, which is positive because as the model performs well during training, we expect the NIST metrics to improve. The efficiency in this experiment looks more

stable compared to the previous experiments (in the range of 10% to 30% of excluded lowest-quality images), and moreover, it manages to reach a value of 0.35, which is the highest among the presented experiments. Given the metrics, the algorithm works as intended, and discarding low-quality images improves its performance.

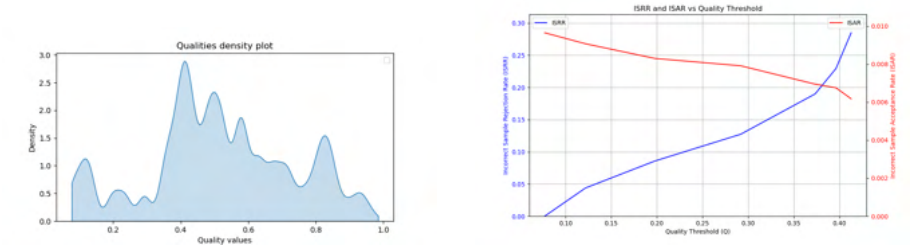


Figure 6.28: Qualities, ISAR, ISRR for Bootstrapping experiment.

As presented in Figure 6.28 on the left, the quality density plot shows the distribution of image qualities with a peak around 0.4 to 0.6, indicating that many images are evaluated as having average quality. There are only a small number of images where the algorithm predicts higher values, ranging from 0.8 to 1. Although the IJB-C dataset contains many images with occlusion, it still includes good-quality images, so the predictions are not always perfect. The ISRR-ISAR plot on the right defines the optimal threshold near 0.37, with ISRR around 0.2 and ISAR at 0.007. It is notable that rejecting 20% of images, based on the optimal intersection point, is quite high. These exact values will be compared with other metrics in the final section of the chapter, as the comparison provides better insights into each model's performance.

The quality scores depicted in Figure 6.29 appear consistent for images with similar characteristics. We can summarize them as follows. High scores are assigned to:

- Properly lit images.
- Images with good resolution, clear details, and sharpness.
- Images where the person is facing the camera directly.

On the other hand, lower scores are given to:

- Blurry images, those with noise or artifacts.
- Images with low light or overexposure.
- Images where the person has a significant head turn.

Overall, the assignment of qualities with the bootstrapping algorithm has much in common with the previous successful algorithms and the initial training. It corresponds to the visual human assessment in most cases, but there are some outlier images present as well.

Figure 6.29: Images sorted by quality for Bootstrapping experiment.

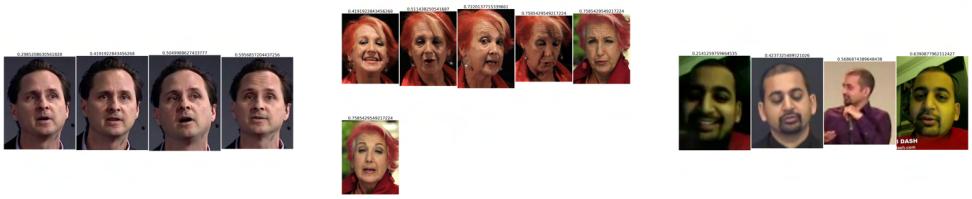


Figure 6.30: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the Bootstrapping experiment.

We refer to Figure 6.30 for the individual qualities. For the individual in first plot, the predicted values are quite consistent compared to other experiments, where the model predicted values smaller than 0.2 and larger than 0.8 for images within the same video frame. Most algorithms predicting qualities measure features such as mouth open or eyes open. In this case, the model effectively captured these details. Additionally, the last image is the closest to a frontal view. For the individual in second plot, the last two images have better resolution. The first four images depict the person in action, with different rotations of the face and more pronounced emotions. For the individual in last plot, the predictions show a gradual improvement. The first image has low resolution and is underexposed. The second image also has low resolution. In the third image, the face is positioned too far away and turned to the side. The last image has the best resolution among those compared.

6.5 FaRL + MLP

In this experiment, we used exactly the same triplets as in the baseline experiment. We extracted the feature vectors using FaRL and trained an MLP instead of CNN-FQ as described in Section 4.3.

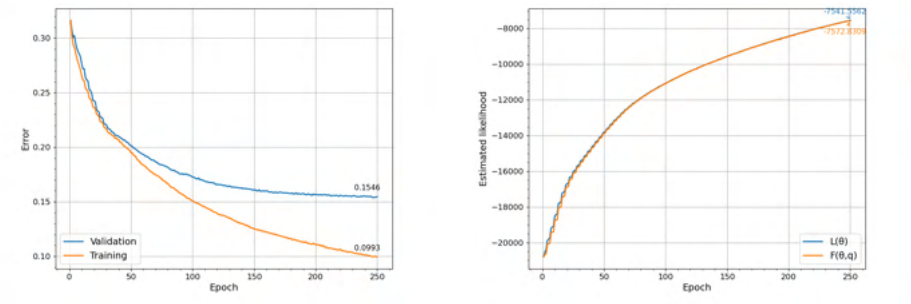


Figure 6.31: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for FaRL experiment.

Examining the outcome of this approach (Figure 6.31) shows a result that is visibly different from the previous experiments. Even though the training error has a value of 9.93%, which is slightly higher, the validation error decreases to 15.46% - one of the best across the experiments described so far. The behavior of the log-likelihood and the auxiliary function is also different: the function F gets very close to L, indicating that the parameters are estimated well.

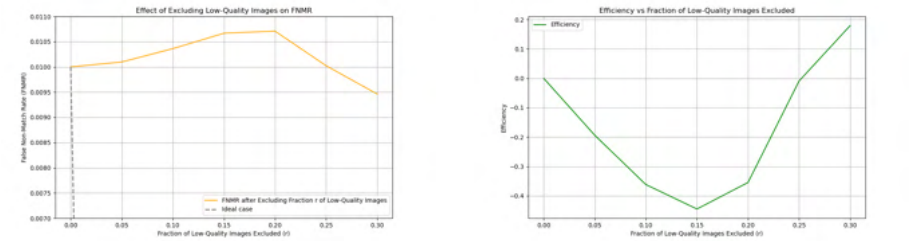


Figure 6.32: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for FaRL experiment.

Figure 6.32 presents the FNMR and efficiency metrics for the FaRL experiment. The FNMR plot implies that excluding more low-quality images does not consistently benefit the model, but we see a larger decrease after a fraction of 0.2. At the same time, the efficiency grows significantly from the same point, 0.2 as well. FNMR manages to decrease from 0.01 to around 0.0095, and the efficiency grows from 0 to around 0.2.

Figure 6.34: Images sorted by quality for FaRL experiment.

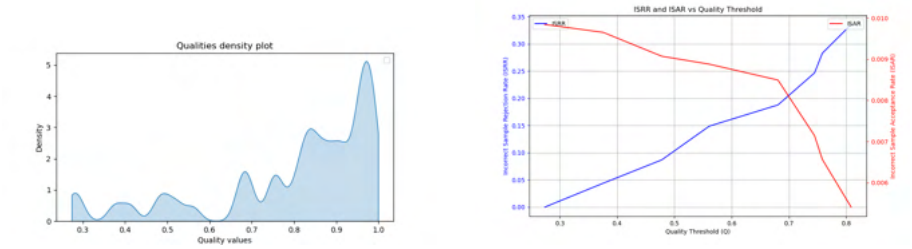


Figure 6.33: Qualities, ISAR, ISRR for FaRL experiment.

Figure 6.33 illustrates the predicted quality scores and ISAR-ISRR plot. The quality density plot shows a similar to Baseline experiment distribution, skewed to higher values, with the difference of starting predicting values from the values around 0.2, meaning that the model does not assign the lowest possible, zero values. Another difference is the higher variability of predicted values, which is a positive aspect. There is a significant peak from 0.8 to 1, indicating that the algorithm tends to give high values to most of the images. Another difference is the higher variability of predicted values, which is a positive aspect. There are peaks around 0.2, 0.5 and 0.7 as well. The quality threshold suggested as optimal by ISAR-ISRR is the highest among all experiments. Comparing the value of ISRR, which is around 0.2 here, to 0.12 in the baseline experiment, shows that the current model is more conservative and leads to more incorrect rejections, which is not a very positive sign. It is important to balance incorrect rejections and acceptances. However, since we are more interested in lower ISRR, the choice of the quality threshold in a real-world setting may be different.

In general, the model can distinguish between different quality levels, as demonstrated in Figure 6.34. Many of the low-quality images have issues such as blurriness, low resolution, poor lighting, or non-frontal poses. Higher-quality images tend to have good lighting, sharpness, and frontal poses. The model appears consistent in assigning lower scores to images with obvious quality issues and higher scores to images without such issues. However, there are a few cases where the quality scores might not align perfectly with visual inspection, but none of the algorithms resulted in perfect predictions for all the images.

Figure 6.35: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the FaRL experiment.

We refer to Figure 6.35 for details on the individual qualities. For Person a), the ordering looks efficient, but the difference between the qualities of the first and second images (which are almost identical) is quite large, likely due to the yaw angle and the reflection on the glasses. For Person b), the first two images are of low resolution, the third one is of higher resolution but has the top part of the face cropped, and the last image shows high resolution with the face closer to the camera. For Person c), the face with the neutral expression got the highest score. However, the quality values are expected to be closer to each other as they stem from the same video.

6.6 Bootstrapping + FaRL

We used triplets similar to those in the initial bootstrapping experiment. Since the seed was set, the initial training triplets are identical. We modified the bootstrapping to work with feature vectors extracted from images and used a multilayer perceptron instead of CNN-FQ. All other settings, such as evaluation and data addition criteria, remain the same as in the original bootstrapping experiment. The validation set is fixed to ensure consistency over epochs and accurate representation.

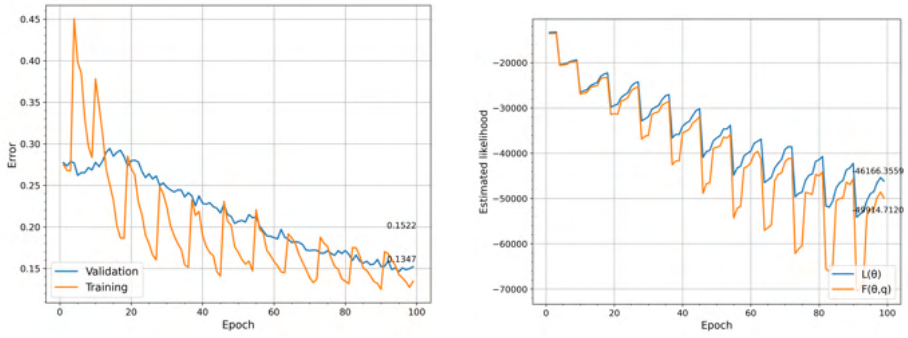


Figure 6.36: The left figure shows the evaluation of the training and validation error as the function of the number of epoch of the EM-algorithm. The right figure visualizes the log-likelihood L and its lower bound optimized by the EM algorithm for Bootstrapping+FaRL experiment.

As demonstrated in Figure 6.36, both the training and validation errors start at a lower level, around 0.3, compared to experiments with image data, where they started around 0.4. The training error, although higher than in CNN-FQ experiments, converges to 15.22%. The validation error surpasses all previous results, achieving a value of 13.47%, which was our desired outcome. The fluctuations in F and L are due to the bootstrapping method itself, but overall, both the log-likelihood and the auxiliary function show growth, with F remaining less than or equal to L.

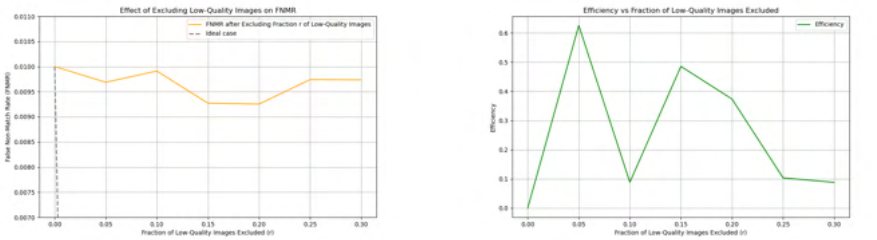


Figure 6.37: The left figure shows the FNMR (y-axis) versus the fraction of the excluded queries (x-axis). The right figure shows the efficiency (y-axis) as the fraction of the excluded queries (x-axis) for Bootstrapping+FaRL experiment.

FNMR and efficiency are presented in Figure 6.37. In comparison with the FaRL experiment on IJB-C and without bootstrapping, the model exhibits similar fluctuative behaviour. However, it shows a slightly higher FNMR and does not display as strong a decreasing trend. The same is true for efficiency; for some reason, it does not perform as well as the initial FaRL. Compared to normal bootstrapping using image data, the model fluctuates with no clear trend. Overall, the method is functional but not ideal. Despite the satisfactory performance during training, its effectiveness is reduced in this context. The vector representation probably lack certain features, resulting in slightly worse performance compared to the image and CNN-based approach.

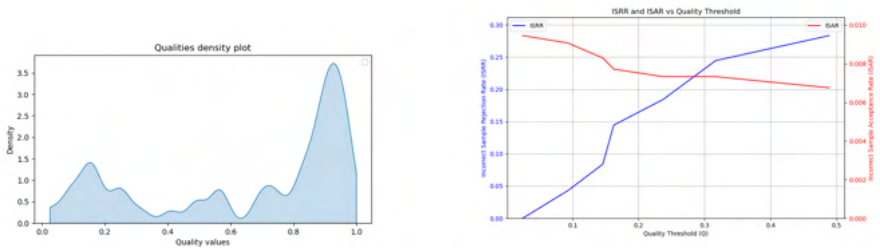


Figure 6.38: Qualities, ISAR, ISRR for Bootstrapping+FaRL experiment.

The quality distribution (Figure 6.38, on the left) in this experiment is very close to that in the original experiment, which is logical since the data representation is the same—the feature vectors—although the dataset size is larger. When compared with bootstrapping on image data, most qualities were around 0.4-0.6, whereas in this experiment, the qualities are higher, skewed to the right with a peak between 0.8 and 1, and another peak around lower values. The ISRR and ISAR values remain similar to those in bootstrapping and FaRL+MLP experiments, but the quality threshold is lower.

Figure 6.39: Images sorted by quality for Bootstrapping+FaRL experiment.

As the common features of low-quality and high-quality images have been previously discussed, we will not mention them once again here. Generally, the quality assignment and ordering (shown in Figure 6.39) in this experiment are consistent with those observed in successful experiments. The model effectively captures features indicative of low-quality images, such as noise and blurriness, and performs similarly for higher-quality images. However, there are still some impostor images when viewed from a human perspective.

Figure 6.40: The examples of input faces for individuals and the predicted scores returned by the CNN-FQ for the FaRL experiment.

Moving on to individual cases presented in Figure 6.40 may provide more details about how the model works. The plot (a) presents a generally controversial sequence of images. In the first and second images, the person is looking down with eyes closed. In the third and fourth images, the person is looking to the side, and in the fourth image, the person is smiling. The fifth image is an impostor; it is visibly the worst among all images but somehow received a better score. The last image visually appears better than the others because the face is almost neutral and almost facing the camera. In plot (b), the ordering of images clearly makes sense, as the quality improves with each subsequent image. The first image is very pixelated, with eyes closed, mouth open, and facing the side. The second image has low resolution. The third image is good, but the hair occludes the top of the face, and the head position is not as straight as in the last image, which has the best resolution. In the final image, the person looks directly at the camera, and despite the occlusion by the microphone, it still visibly looks as the best one. In plot (c), the first image shows emotions, is blurred, and is facing to the side. The second and third images are non-frontal and less sharp. However, the second and third images could probably be swapped. The last image has better lighting and resolution and is closer to a frontal view.

6.7 Discussion of results

It is crucial to explore and analyze the metrics together. Each metric (such as validation error or false non-match rate) is defined over a set of different parameters and may not reflect the whole underlying complexity of a problem. It describes the performance only partly, and one metric can reveal insights that another might overlook. Moreover, in real-world applications, we usually have more than one specific goal.

In this thesis we used a range of metrics each evaluating a specific aspect of the face quality predictor. Namely, we used the following metrics:

- Triplet classification error show how well the model can predict the correct ordering of the triplets. Training error indicates how well the model fits the training data, while validation error shows how well the model generalizes to unseen data. The triplet classification error is a proxy objective which is easy to optimize based on the available data and

their annotations. However, the true metrics to evaluate the performance of the quality predictor, like FNMR, are shown below in the list.

- FNMR and Efficiency show how discarding the faces with predicted low quality improve the performance of the underlying face verification system. We report FNMR after discarding 1 to 30 percent of images. FNMR below 0.01 indicates improvement.
- The distribution of quality score predicted from the test faces is important, especially in combination with knowledge about the data. For example, if we know that the dataset is diverse, but the model predicts only high or low values, it may signal that something is not working as intended.
- ISRR versus ISAR: balancing these metrics is crucial because a high threshold increases ISRR, leading to more frequent denial of access to authorized users. On the other hand, a low threshold reduces the ISRR, but it can increase the risk of unauthorized access described by ISAR.
- Visual quality assessment: the goal is to visualize how the quality predicted by the trained model correlates with the human perceived quality.

In the summary tables we report the results for the following methods:

- Baseline: implementation of the original algorithm of [47].
- Unique triplet generation: experiment with unique triplets.
- Aligned: experiment on aligned images.
- Aligned Unique: experiment on aligned images with unique triplets.
- Bootstrapping: bootstrapping with the best chosen configuration.
- FaRL: experiment with feature vectors and a MLP.
- Bootstrapping+FaRL: bootstrapping on joint IJB-C and CASIA dataset using feature vectors instead of images.

Table 6.2: The triplet classification error evaluated on the training and testing data shown for the evaluated methods, sorted from best by validation error

Experiment	Training error	Validation error
Bootstrapping+FaRL	15.22%	13.47%
Bootstrapping	6.28%	14.79%
FaRL	9.93%	15.46%
Baseline	5.75%	21.5%
Aligned	8.95%	22.45%
Aligned Unique	3.58%	45.71%
Unique triplet generation	3.4%	46.47%

Table 6.2 summarizes training and validation triplet classification error for all the compared methods. It is evident that the models are generally capable of fitting well to the training data. The validation error, however, provides insight into how the model performs with unseen data, making it a crucial metric for evaluation. The best result is achieved with the Bootstrapping+FaRL model, where a large dataset of images from IJB-C and CASIA was used, along with feature vectors and a different neural network architecture from CNN-FQ. The second-best result is from the initial bootstrapping, which had a slightly higher validation error but achieved half the training error in comparison with the Bootstrapping+FaRL. The FaRL experiment, which was conducted on the same data as the Baseline, outperformed the Baseline experiment by reducing the error by an additional 6%, even with a much simpler architecture, probably thanks to the fact that FaRL was trained on a dataset of around 20 million images [50] and is capable of extracting really low-level but important details from the face images. The Baseline and Aligned experiments yielded similar errors, indicating that the alignment process did not significantly benefit the model. The Unique triplet generation and Aligned Unique experiments performed poorly, likely due to insufficient dataset sizes leading to overfitting on new data.

Table 6.3: FNMR and Efficiency computed on the test set. FNMR below 0.01 indicates improvement due to using the face quality improvement.

Experiment	FNMR(30)	Efficiency(30)
Bootstrapping	0.00895	0.349
FaRL	0.00946	0.179
Baseline	0.00947	0.175
Unique triplet generation	0.00974	0.086
Bootstrapping+FaRL	0.00973	0.084
Aligned	0.00999	0.002
Aligned Unique	0.0105	-0.185

Table 6.3 shows the FNMR and efficiency values when the highest percentage of data, 30%, is excluded. Although the differences overall are not very significant, some observations and conclusions can still be made. The lowest false non-match rate and highest efficiency were achieved in the Bootstrapping experiment. FaRL is the second-best experiment. The results of the Baseline and FaRL experiments are similar, which is logical since they were run on the same data. However, we would expect FaRL to outperform the Baseline more significantly, as observed when comparing the validation error. Surprisingly, the results of the Unique triplet generation and the Bootstrapping+FaRL experiments are close to each other. However, it should be noted that the NIST metrics for the experiment Unique triplet generation did not follow any clear trend, so this might be a coincidence. The NIST metrics on aligned data (Aligned, Aligned Unique) are the least successful.

Table 6.4: ISAR, ISRR and quality thresholds summary for the experiments

Experiment	Threshold	ISRR	ISAR
Baseline	0.35	0.12	0.008
Unique triplet generation	0.10	0.12	0.008
Aligned	0.15	0.14	0.0081
Aligned Unique	0.003	0.11	0.0082
Bootstrapping	0.37	0.20	0.007
FaRL	0.70	0.20	0.12
Bootstrapping+FaRL	0.28	0.22	0.007

Table 6.4 summarizes the ISAR, ISRR, and quality thresholds for the various experiments. ISAR and ISRR values are from the optimal point—the point of intersection of the two error lines. The key observations are:

- The quality threshold is the highest for the FaRL experiment, with a value of 0.7, which is significantly higher than the thresholds for the other experiments. This indicates that the model is very conservative. On the other hand, it is important to remember that the FaRL experiment tended to predict very high qualities for face images in general.
- The highest values of ISRR are for the Bootstrapping, FaRL, and Bootstrapping+FaRL experiments. We generally prefer the lowest possible score, and a value of around 0.2 is relatively high. If we wanted to ensure the lowest possible ISRR value, we would have to sacrifice situations where the ISAR—incorrect acceptance situations—is higher.
- The ISAR values are very close for all the experiments except FaRL, even given the different quality thresholds. This suggests that the models have a similar ability to correctly identify good and bad quality images, but they set different thresholds for classification. This can be explained by the fact that the algorithms predict different quality values, which can be seen from the quality density plots presented for every algorithm.

Table 6.5: Validation error, FNMR and the number of training faces (images) for the experiments, sorted from largest number of faces.

Experiment	Validation error	FNMR	Num. of train faces
Bootstrapping	14.79%	0.00895	158,000
Bootstrapping+FaRL	13.47%	0.00973	142,000
Aligned	22.45%	0.00999	61,000
FaRL	15.46%	0.00946	55,500
Baseline	21.5%	0.00947	55,500
Unique triplet generation	46.47%	0.00974	21,600
Aligned Unique	45.71%	0.0105	21,500

Table 6.5 presents the experiments with a focus on the number of faces (images) used during the training, along with the corresponding validation error and FNMR. This analysis aims to highlight the impact of the number of images on the performance metrics. Bootstrapping had one of the lowest validation errors and the best (lowest) false non-match rate, probably due to the largest number of images used in the training process. Bootstrapping+FaRL, which also used a large dataset, achieved the best validation error among all experiments. However, its FNMR was higher, and its training error was quite high as well, indicating some trade-offs in performance metrics. The Aligned experiment shows that a larger dataset size is not always the key to success. Despite having a slightly larger dataset, it performed worse than the Baseline experiment. This could be due to issues during the alignment and cropping process or a smaller percentage of non-repeating images in this dataset. FaRL and Baseline were trained on the same data, resulting in very close FNMR values. However, the validation error for FaRL was visibly lower, suggesting that the vector representations used in FaRL might provide valuable information. The experiments with the smallest number of images, Unique triplet generation and Aligned Unique, aimed to use the maximum number of unique images possible. The number of triplets in these experiments was significantly smaller than in other datasets, making it insufficient for the model to learn effectively. Additionally, there could have been other underlying issues in these datasets that impacted their performance.

When addressing the visual quality assessment, the general qualities prediction and ordering worked satisfactorily for most of the experiments. For Unique triplet generation and Aligned Unique, it worked poorly. The qualities for individuals generally worked for all models, but it is important to note that overall their effectiveness highly depends on the specific case.

As previously discussed, the selection of the best method requires a comprehensive evaluation of all metrics together. Overall, the most promising results were obtained from the FaRL, Bootstrapping, and their combined approach, Bootstrapping+FaRL. Additional data preprocessing steps, such as alignment, did not significantly change the outcomes. One possible explanation is that the images were already normalized to some extent. Another hypothesis is that the aligned images may have lost some crucial information due to differences in cropping and bounding box configurations. Unfortunately, the experiments with unique triplets did not yield successful results. However, there is potential for improvement with an increased number of triplets or more sophisticated implementation strategy. Regarding Bootstrapping, our hypothesis was confirmed; the larger dataset indeed enhanced performance, which is a positive sign. FaRL and FaRL+Bootstrapping were notably successful, moreover, the utilization of vector representations significantly accelerated the training process.

Chapter 7
Conclusion

In this thesis, we have explored the topic of face image quality prediction. We have presented the theoretical background, including various definitions and aspects of face image quality, factors affecting it, use cases, and approaches to its assessment.

The goals and contributions of this work were to develop an efficient implementation of the algorithm to learn the face image quality predictor [47] and to explore different strategies for generating training triplets. We also aimed to implement transfer learning using image representations from deep neural models pre-trained on large datasets, specifically using FaRL trained on a 20M database [50]. Lastly, we aimed to evaluate all implemented methods using the metrics defined by the ongoing NIST challenge on face image quality assessment [13].

In the practical part of the work, we introduced the baseline experiment, mirroring the implementation of the original method [47], which was subsequently enhanced in various ways. We explored different approaches as stated in the goals. We implemented an experiment with additional data processing and alignment. We attempted to generate a more diverse dataset with a high percentage of unique faces in triplets. We also implemented bootstrapping using joint CASIA+IJBC dataset, an experiment with FaRL-extracted vectors on the same data as the baseline experiment, and finally a combination of FaRL and bootstrapping. We presented, described, and implemented the metrics proposed by NIST to evaluate all the methods. Each experiment was evaluated using these NIST metrics along with a discussion. We provided chosen images with quality predictions to ensure that, with respect to human perception, the method works as well. In general, we had several hypotheses on how to improve the implementation and what to try out. We have summarized all the experiment results in Section 6.7 and discussed which enhancements were successful and which were not.

The main finding was that a larger dataset and working with different image representations, such as feature vectors, were indeed beneficial. From the performed experiments, we can conclude that the goals of the thesis were accomplished. We tried out different improvements to the algorithm, evaluated them using real-world NIST metrics, and identified the most promising approaches.

This thesis opens up opportunities for future research on the topic. There are several possible paths for enhancing and extending this work:

- The multilayer perceptron architecture used in the FaRL experiment was relatively simple. Exploring improvements to the MLP architecture working with FaRL could yield better results. A more sophisticated architecture might lead to even greater improvements.
- We observed a significant increase in training speed when working with feature vectors instead of images. Another potential direction could involve training FaRL on an even larger dataset by using more triplets from the joint IJB-C+CASIA dataset or by incorporating new datasets.
- Another possibility is to explore further enhancements of the current algorithm. This could involve experimenting with efficient data generation strategies or trying different backbones other than the already used CNN-FQ and FaRL.

Bibliography

[1] Gaurav Aggarwal, Soma Biswas, Patrick J Flynn, and Kevin W Bowyer. Predicting performance of face recognition systems: An image characterization approach. In *CVPR 2011 WORKSHOPS*, pages 52–59. IEEE, 2011.

[2] Fernando Alonso-Fernandez, Julian Fierrez, and Javier Ortega-Garcia. Quality measures in biometric systems. *IEEE Security & Privacy*, 10(6):52–62, 2011.

[3] Lacey Best-Rowden and Anil K. Jain. Learning face image quality from human assessments. *IEEE Transactions on Information Forensics and Security*, 13(12):3064–3077, 2018.

[4] J Ross Beveridge, David S Bolme, Bruce A Draper, Geof H Givens, Yui Man Lui, and P Jonathon Phillips. Quantifying how lighting and focus affect face recognition performance. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 74–81. IEEE, 2010.

[5] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. 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, 2018.

[6] Jiansheng Chen, Yu Deng, Gaocheng Bai, and Guangda Su. Face image quality assessment based on learning to rank. *IEEE signal processing letters*, 22(1):90–94, 2014.

[7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[8] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*, 2019.

[9] Stephen Elliott, Eric Kukula, and Nathan Sickler. The challenges of the environment and the human/biometric device interaction on biometric system performance. In *International Workshop on Biometric Technologies-Special forum on Modeling and Simulation in Biometric Technology*, 2004.

[10] Biying Fu, Cong Chen, Olaf Henniger, and Naser Damer. A deep insight into measuring face image utility with general and face-specific image quality metrics. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 905–914, 2022.

[11] Xiufeng Gao, Stan Z Li, Rong Liu, and Peiren Zhang. Standardization of face image sample quality. In *Advances in Biometrics: International Conference, ICB 2007, Seoul, Korea, August 27-29, 2007. Proceedings*, pages 242–251. Springer, 2007.

[12] Patrick Grother, Patrick Grother, Mei Ngan, and Kayee Hanaoka. Face recognition vendor test (frvt) part 2: Identification, 2019.

[13] Patrick Grother, Austin Hom, Mei Ngan, and Kayee Hanaoka. Ongoing face recognition vendor test (frvt)–part 5: Face image quality assessment. *NIST Interagency Report*, 2020.

[14] Patrick Grother and Elham Tabassi. Performance of biometric quality measures. *IEEE transactions on pattern analysis and machine intelligence*, 29(4):531–543, 2007.

[15] Javier Hernandez-Ortega, Javier Galbally, Julian Fierrez, Rudolf Haraksim, and Laurent Beslay. Faceqnet: Quality assessment for face recognition based on deep learning. In *2019 International Conference on Biometrics (ICB)*, pages 1–8. IEEE, 2019.

[16] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[17] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[18] Yuge Huang, Pengcheng Shen, Ying Tai, Shaoxin Li, Xiaoming Liu, Jilin Li, Feiyue Huang, and Rongrong Ji. Improving face recognition from hard samples via distribution distillation loss. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pages 138–154. Springer, 2020.

[19] ISO/IEC JTC1 SC37 Biometrics. Iso/iec 29794-1:2016 information technology - biometric sample quality - part 1: Framework. International Organization for Standardization, 2016.

[20] ISO/IEC JTC1 SC37 Biometrics. Iso/iec 2382-37:2017 information technology – vocabulary – part 37: Biometrics. International Organization for Standardization, 2017.

[21] Hyung-Il Kim, Seung Ho Lee, and Man Ro Yong. Face image assessment learned with objective and relative face image qualities for improved face recognition. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4027–4031. IEEE, 2015.

[22] Emine Krichen, Sonia Garcia-Salicetti, and Bernadette Dorizzi. A new probabilistic iris quality measure for comprehensive noise detection. In *2007 First IEEE International Conference on Biometrics: Theory, Applications, and Systems*, pages 1–6. IEEE, 2007.

[23] Ruggero Donida Labati, Angelo Genovese, Enrique Muñoz, Vincenzo Piuri, Fabio Scotti, and Gianluca Sforza. Biometric recognition in automated border control: a survey. *ACM Computing Surveys (CSUR)*, 49(2):1–39, 2016.

[24] Huitao Luo. A training-based no-reference image quality assessment algorithm. In *2004 International Conference on Image Processing, 2004. ICIP'04.*, volume 5, pages 2973–2976. IEEE, 2004.

[25] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 international conference on biometrics (ICB)*, pages 158–165. IEEE, 2018.

[26] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14225–14234, 2021.

[27] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[28] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42:145–175, 2001.

[29] Javier Ortega-Garcia, Julian Fierrez, Fernando Alonso-Fernandez, Javier Galbally, Manuel R Freire, Joaquin Gonzalez-Rodriguez, Carmen Garcia-Mateo, Jose-Luis Alba-Castro, Elisardo Gonzalez-Agulla, Enrique Otero-Muras, et al. The multiscenario multienvironment biosecure multimodal database (bmdb). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1097–1111, 2009.

[30] Fu-Zhao Ou, Xingyu Chen, Ruixin Zhang, Yuge Huang, Shaoxin Li, Jilin Li, Yong Li, Liujuan Cao, and Yuan-Gen Wang. Sdd-fiqa: unsupervised face image quality assessment with similarity distribution distance. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7670–7679, 2021.

[31] Connor J Parde, Carlos Castillo, Matthew Q Hill, Y Ivette Colon, Swami Sankaranarayanan, Jun-Cheng Chen, and Alice J O'Toole. Deep convolutional neural network features and the original image. *arXiv preprint arXiv:1611.01751*, 2016.

[32] Devi Parikh and Kristen Grauman. Relative attributes. In *2011 International Conference on Computer Vision*, pages 503–510. IEEE, 2011.

[33] M.I. Schlesinger. A connection between learning and self-learning in the pattern recognition (in Russian). *Kibernetika*, 2:81–88, 1968.

[34] Torsten Schlett, Christian Rathgeb, Olaf Henniger, Javier Galbally, Julian Fierrez, and Christoph Busch. Face image quality assessment: A literature survey. *ACM Computing Surveys (CSUR)*, 54(10s):1–49, 2022.

[35] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[36] Michael E Schuckers. *Computational methods in biometric authentication: statistical methods for performance evaluation.* Springer Science & Business Media, 2010.

[37] Harin Sellahewa and Sabah A Jassim. Image-quality-based adaptive face recognition. *IEEE Transactions on Instrumentation and measurement*, 59(4):805–813, 2010.

[38] Yichun Shi and Anil K Jain. Probabilistic face embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6902–6911, 2019.

[39] Elham Tabassi and Patrick Grother. *Biometric Quality:The last 1US Department of Commerce, National Institute of Standards and Technology, 2008.*

[40] Philipp Terhorst, Jan Niklas Kolf, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Ser-fiq: Unsupervised estimation of face image quality based on stochastic embedding robustness. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, *pages 5651–5660, 2020.*

[41] S Vignesh, KVSNL Manasa Priya, and Sumohana S Channappayya. Face image quality assessment for face selection in surveillance video

using convolutional neural networks. In 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 577–581. IEEE, 2015.*

[42] Dayong Wang, Charles Otto, and Anil K Jain. Face search at scale. IEEE transactions on pattern analysis and machine intelligence*, 39(6):1122–1136, 2016.*

[43] Pankaj Wasnik, Kiran B Raja, Raghavendra Ramachandra, and Christoph Busch. Assessing face image quality for smartphone based face recognition system. In 2017 5th International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6. IEEE, 2017.*

[44] Laurenz Wiskott, Jean-Marc Fellous, Nobert Krüger, and Christoph Von Der Malsburg. Face recognition by elastic bunch graph matching. In Intelligent biometric techniques in fingerprint and face recognition*, pages 355–396. Routledge, 2022.*

[45] Yongkang Wong, Shaokang Chen, Sandra Mau, Conrad Sanderson, and Brian C Lovell. Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. In CVPR 2011 WORKSHOPS*, pages 74–81. IEEE, 2011.*

[46] Osamu Yamaguchi, Kazuhiro Fukui, and K-i Maeda. Face recognition using temporal image sequence. In Proceedings third IEEE international conference on automatic face and gesture recognition*, pages 318–323. IEEE, 1998.*

[47] Andrii Yermakov and Vojtech Franc. Cnn based predictor of face image quality. In International Conference on Pattern Recognition*, pages 679–693. Springer, 2021.*

[48] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. arXiv preprint arXiv:1411.7923*, 2014.*

[49] Guangpeng Zhang and Yunhong Wang. Asymmetry-based quality assessment of face images. In International Symposium on Visual Computing*, pages 499–508. Springer, 2009.*

[50] Yinglin Zheng, Hao Yang, Ting Zhang, Jianmin Bao, Dongdong Chen, Yangyu Huang, Lu Yuan, Dong Chen, Ming Zeng, and Fang Wen. General facial representation learning in a visual-linguistic manner. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18697–18709, 2022.*

Appendix A

Used AI Software

In accordance with the guidelines provided in Používání UI ve studijních pracích[1], I declare that I have used the following AI software during the work of this thesis:

- ChatGPT (OpenAI): grammar control, rephrasing. [2]

[1] https://intranet.fel.cvut.cz/cz/rozvoj/MP-pouzivani-ui.pdf
[2] https://chatgpt.com

Appendix B

Images with predicted quality scores

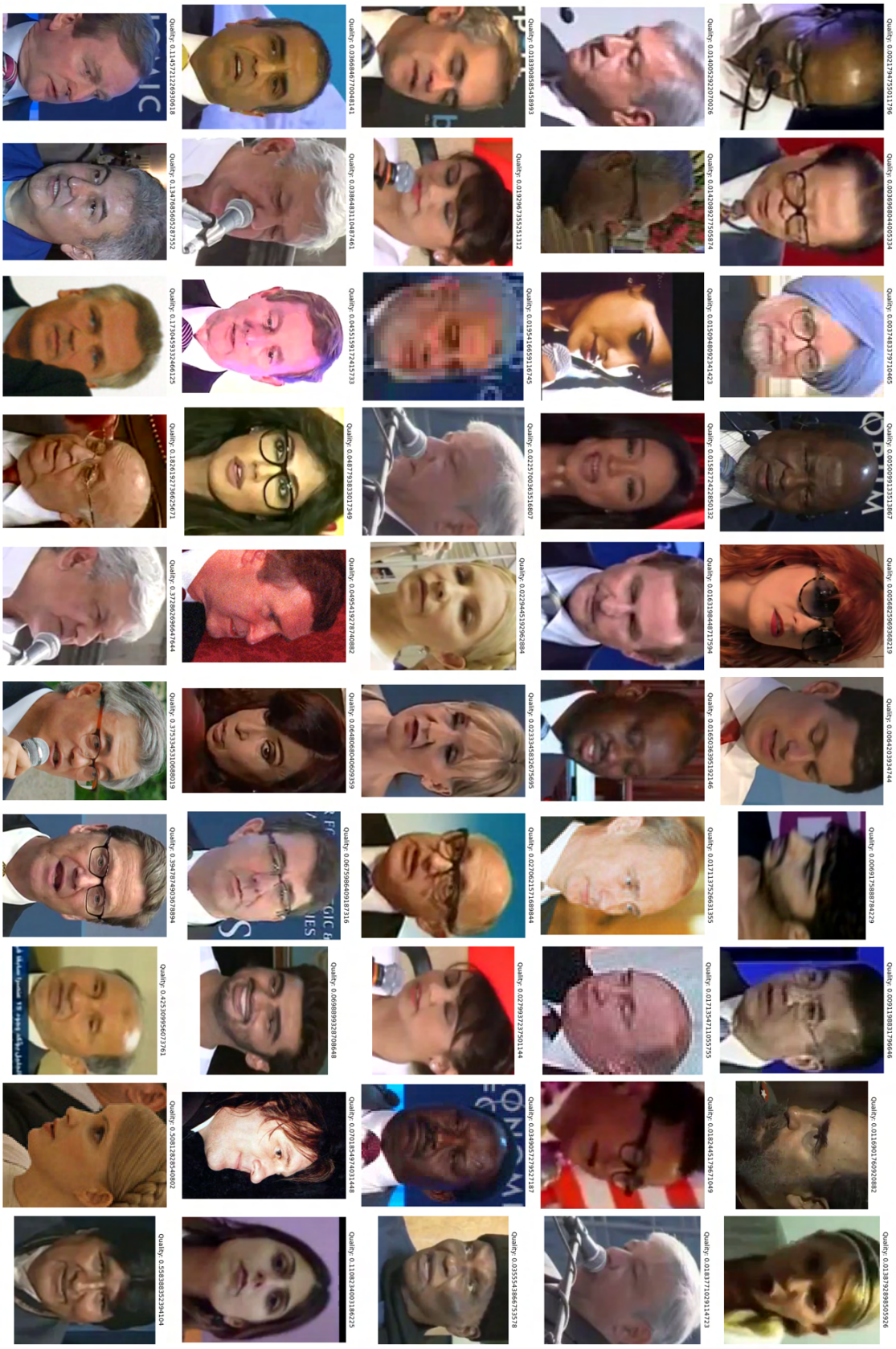


Figure B.1: Images sorted by quality for Baseline experiment.

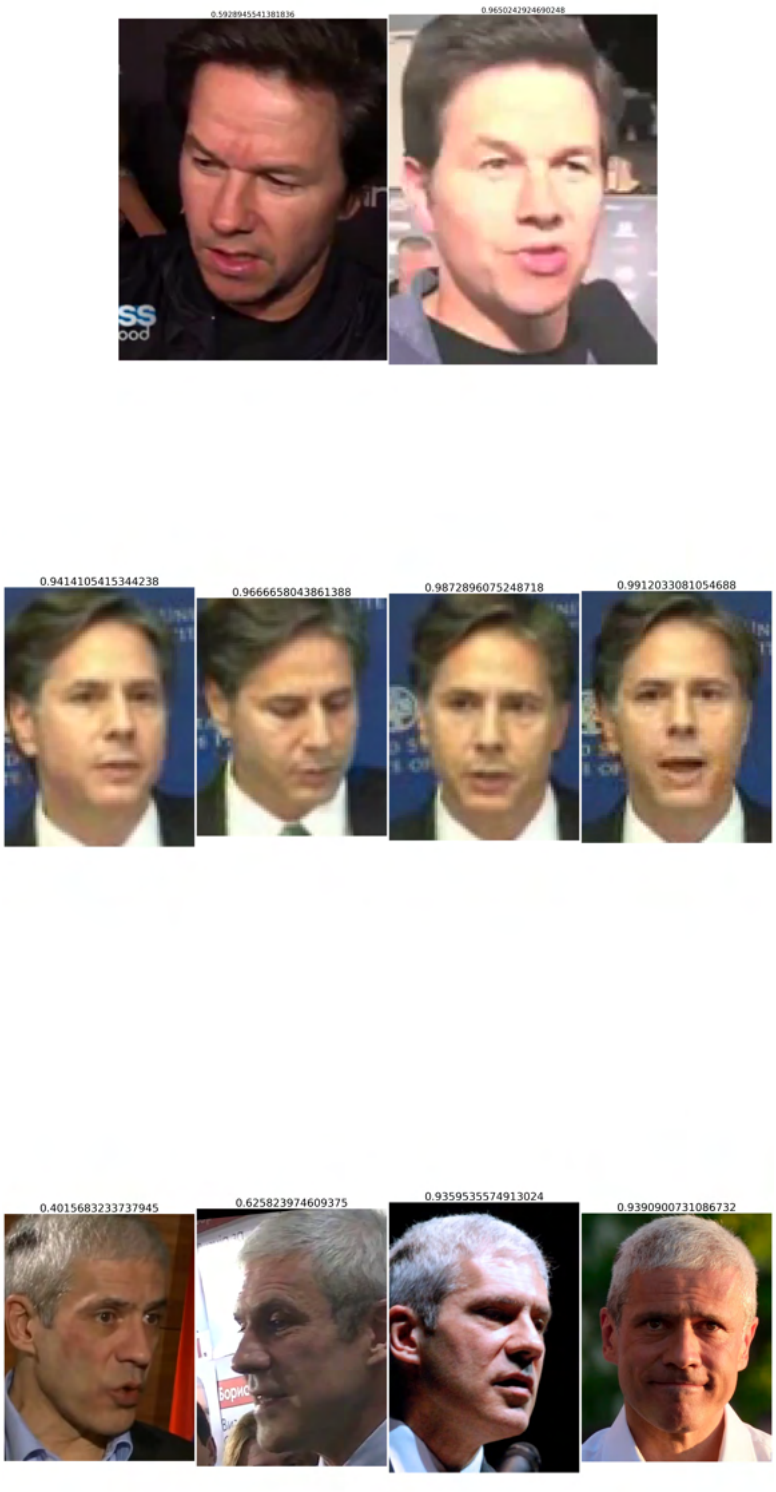


Figure B.2: Qualities for different persons in Baseline experiment.

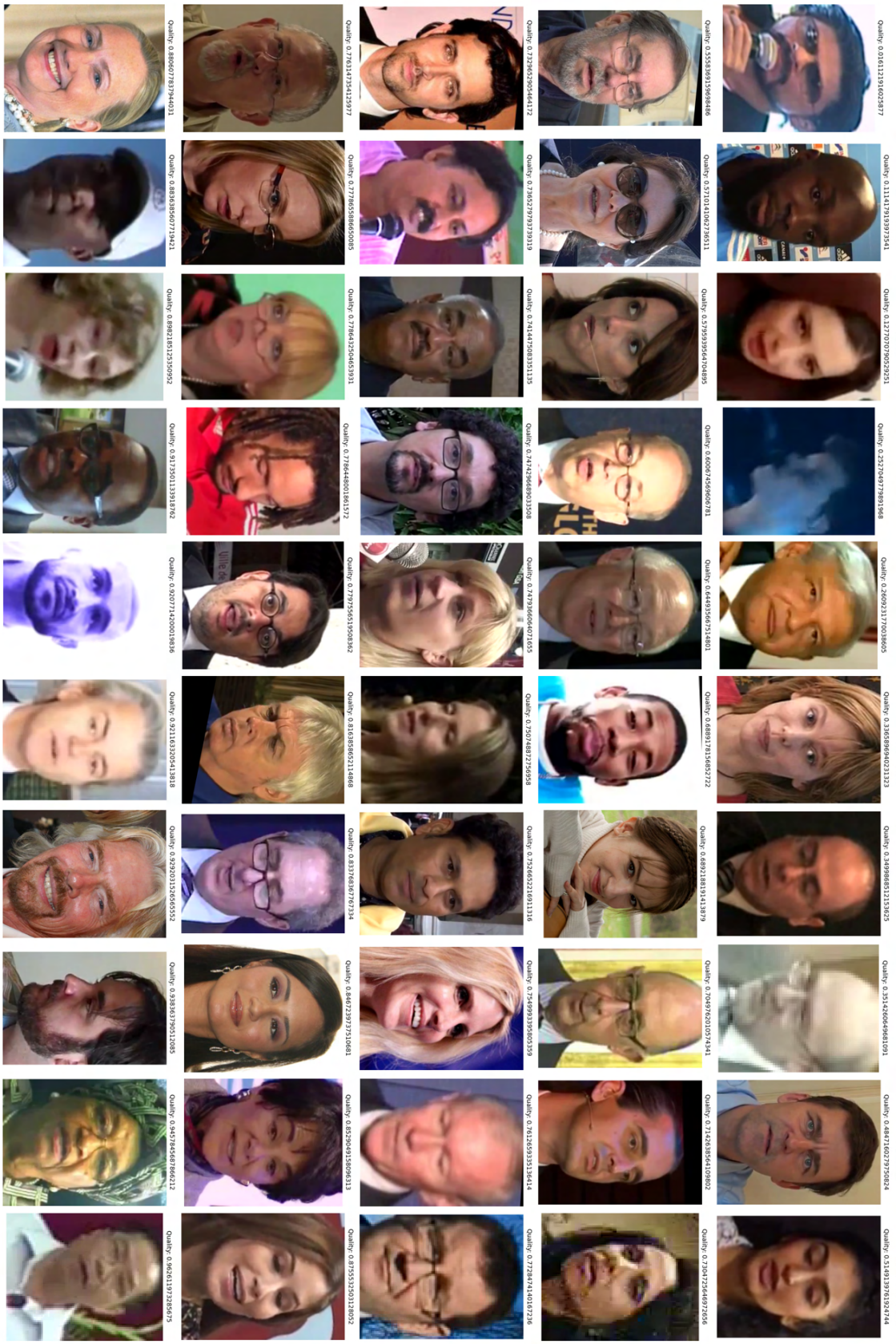


Figure B.3: Images sorted by quality for Unique triplet generation experiment.

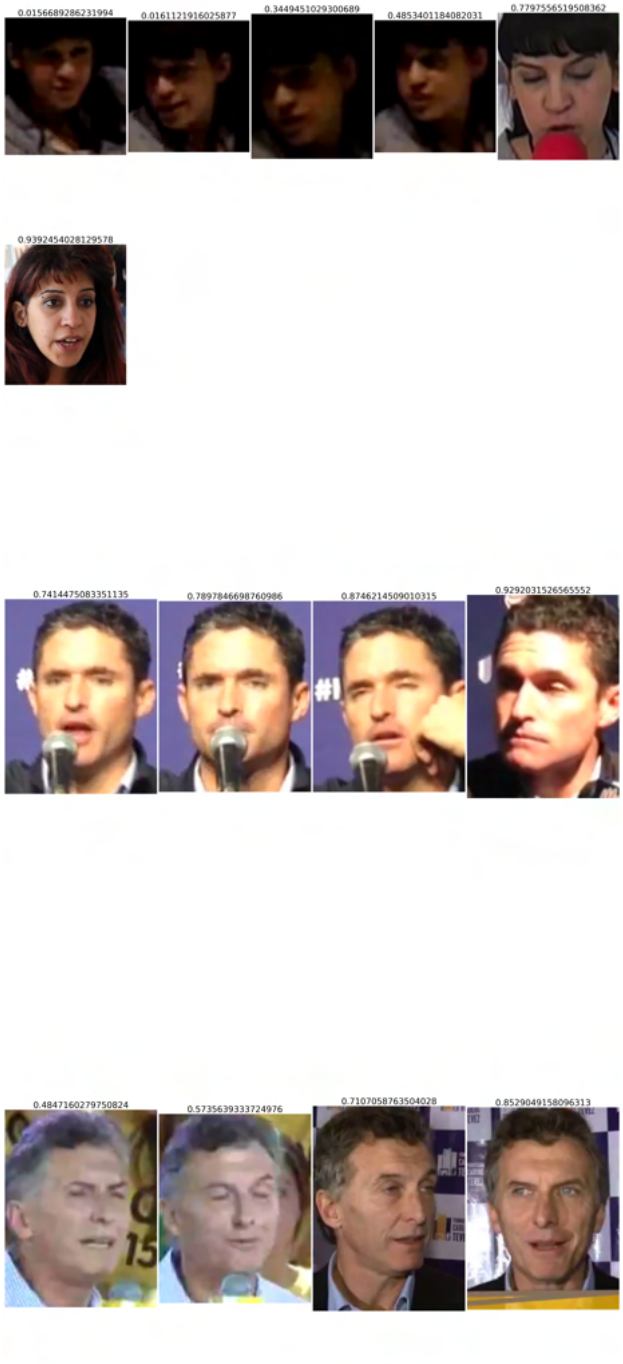


Figure B.4: Qualities for different persons in Unique triplet generation experiment.

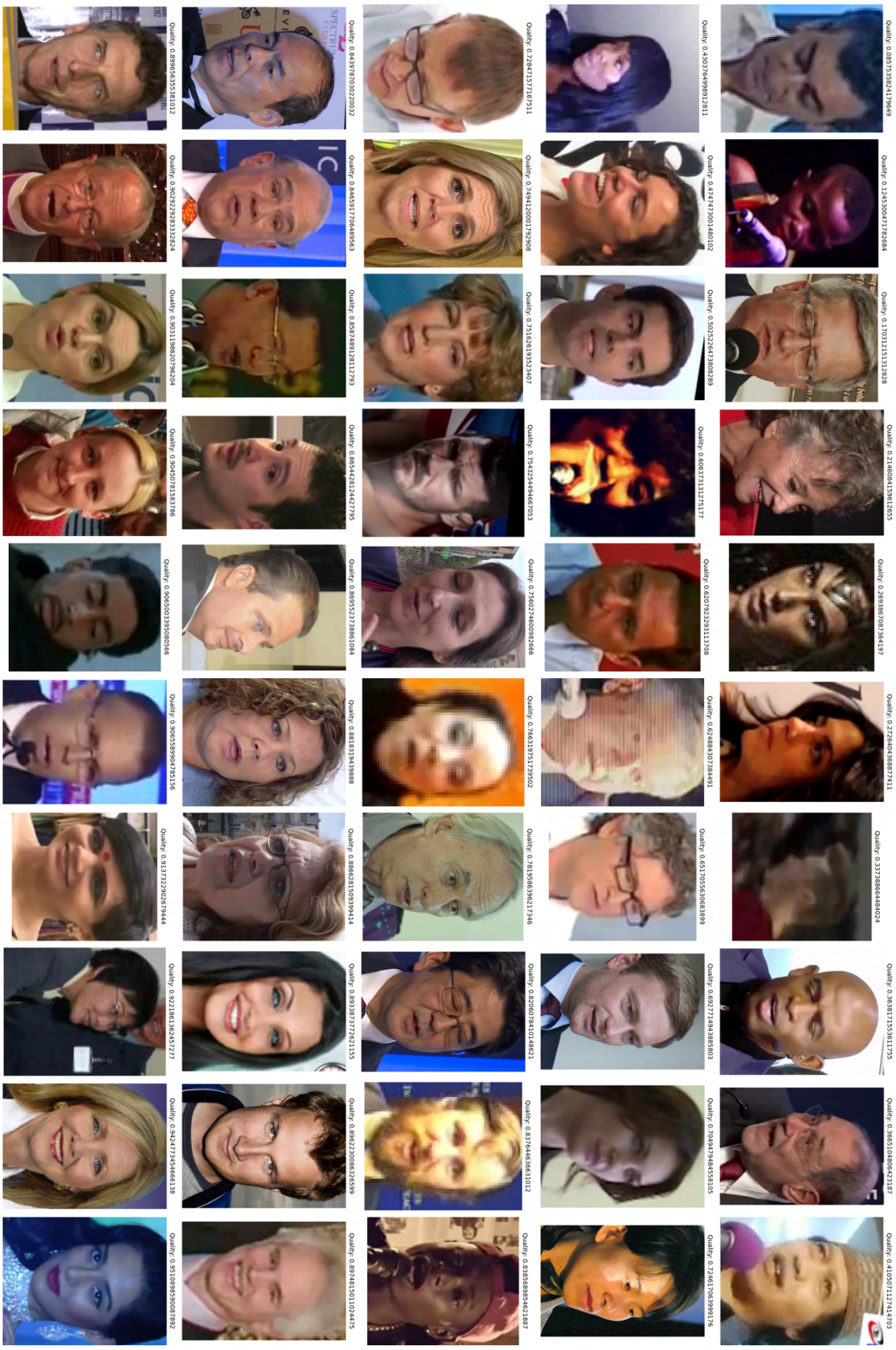


Figure B.5: Images sorted by quality for Aligned experiment.

Figure B.6: Qualities for different persons in Aligned experiment.

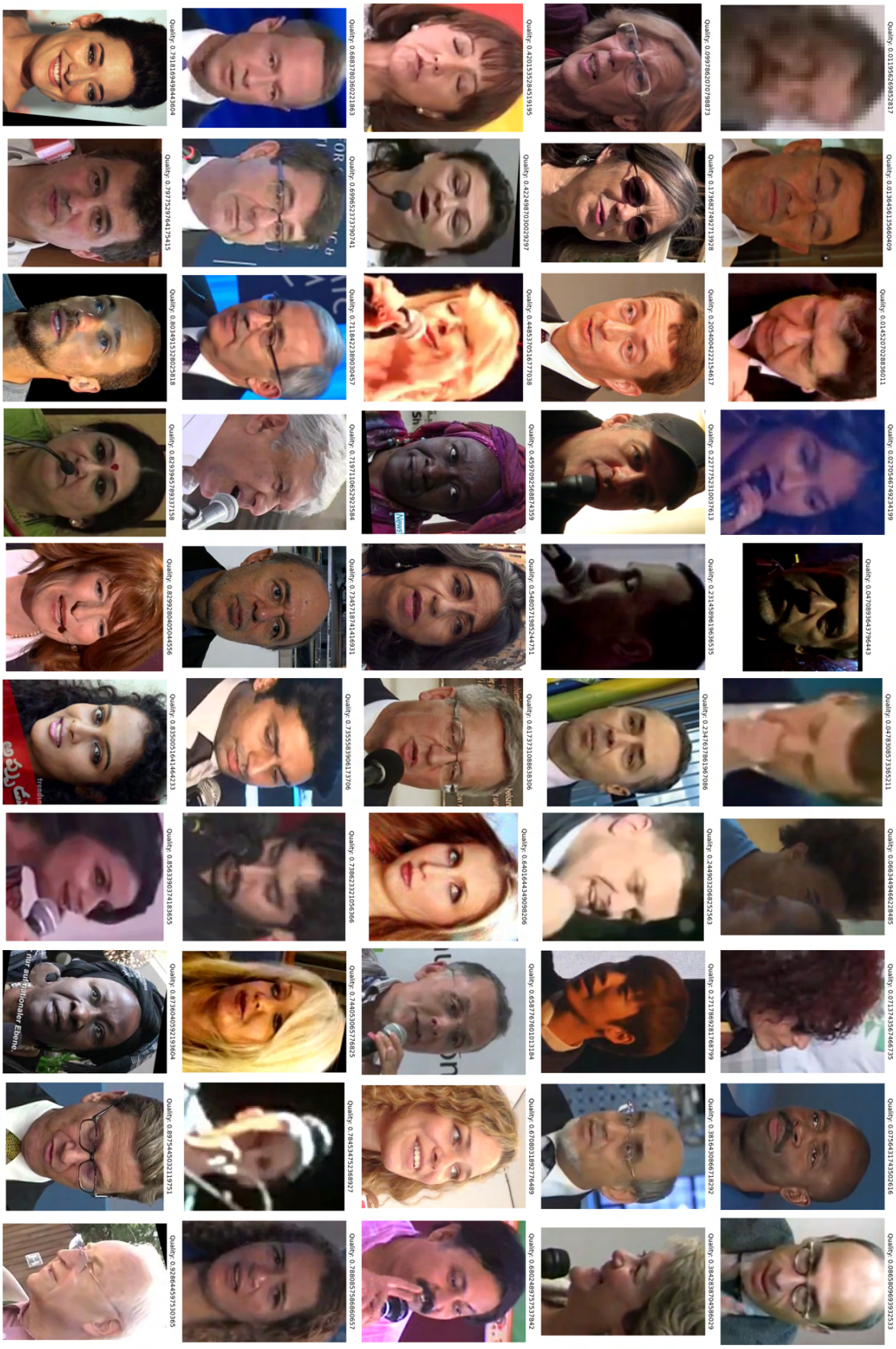


Figure B.7: Images sorted by quality for Aligned Unique experiment.

Figure B.8: Qualities for different persons in Aligned Unique experiment.

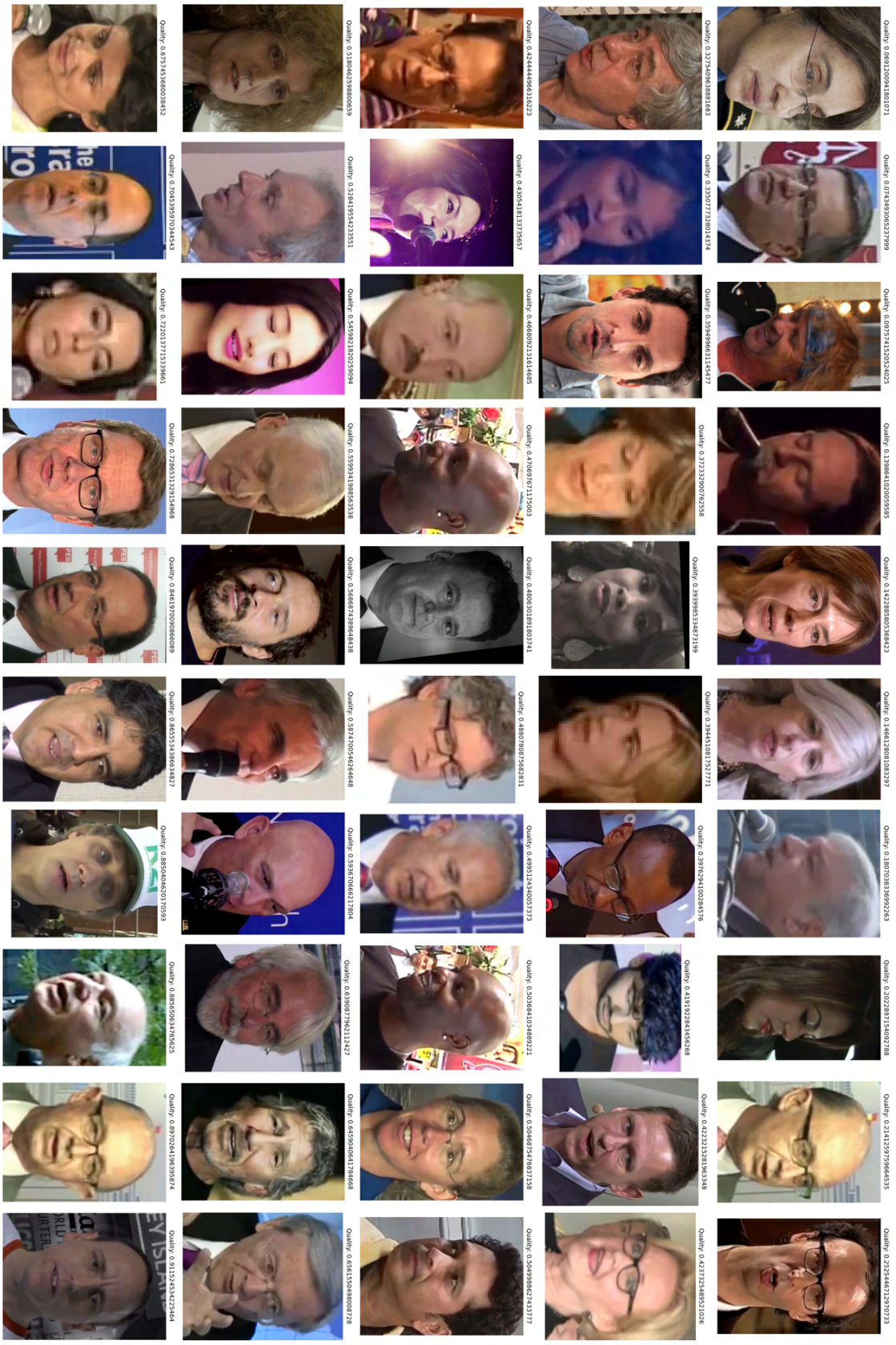


Figure B.9: Images sorted by quality for Bootstrapping experiment.

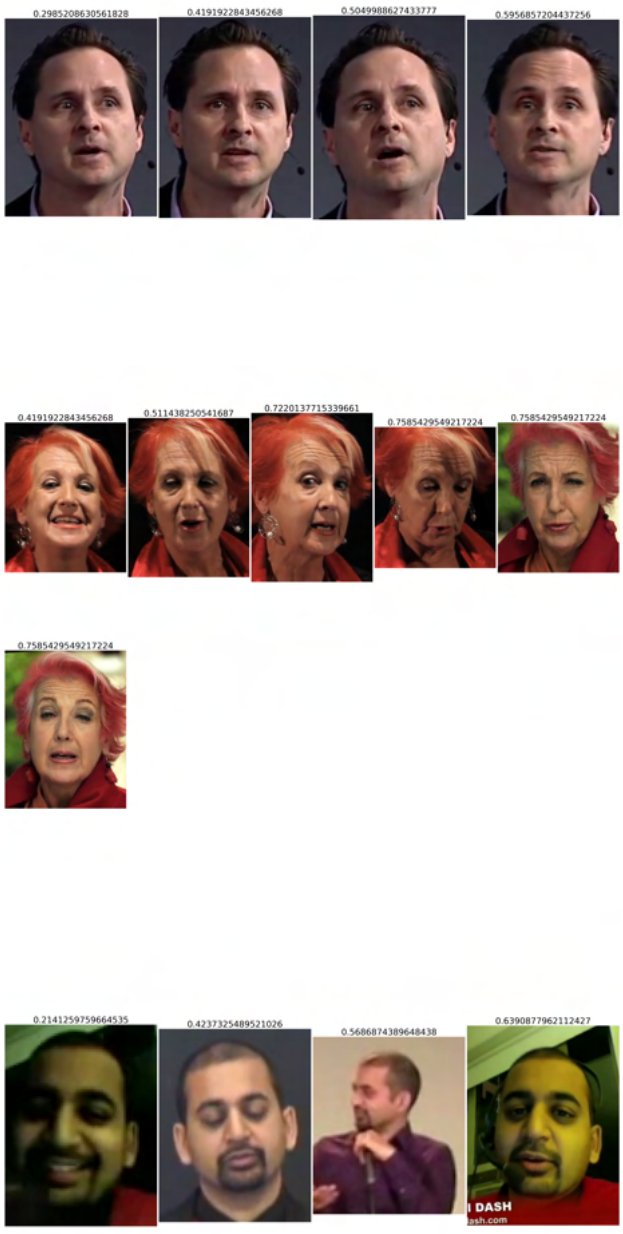


Figure B.10: Qualities for different persons in Bootstrapping experiment.

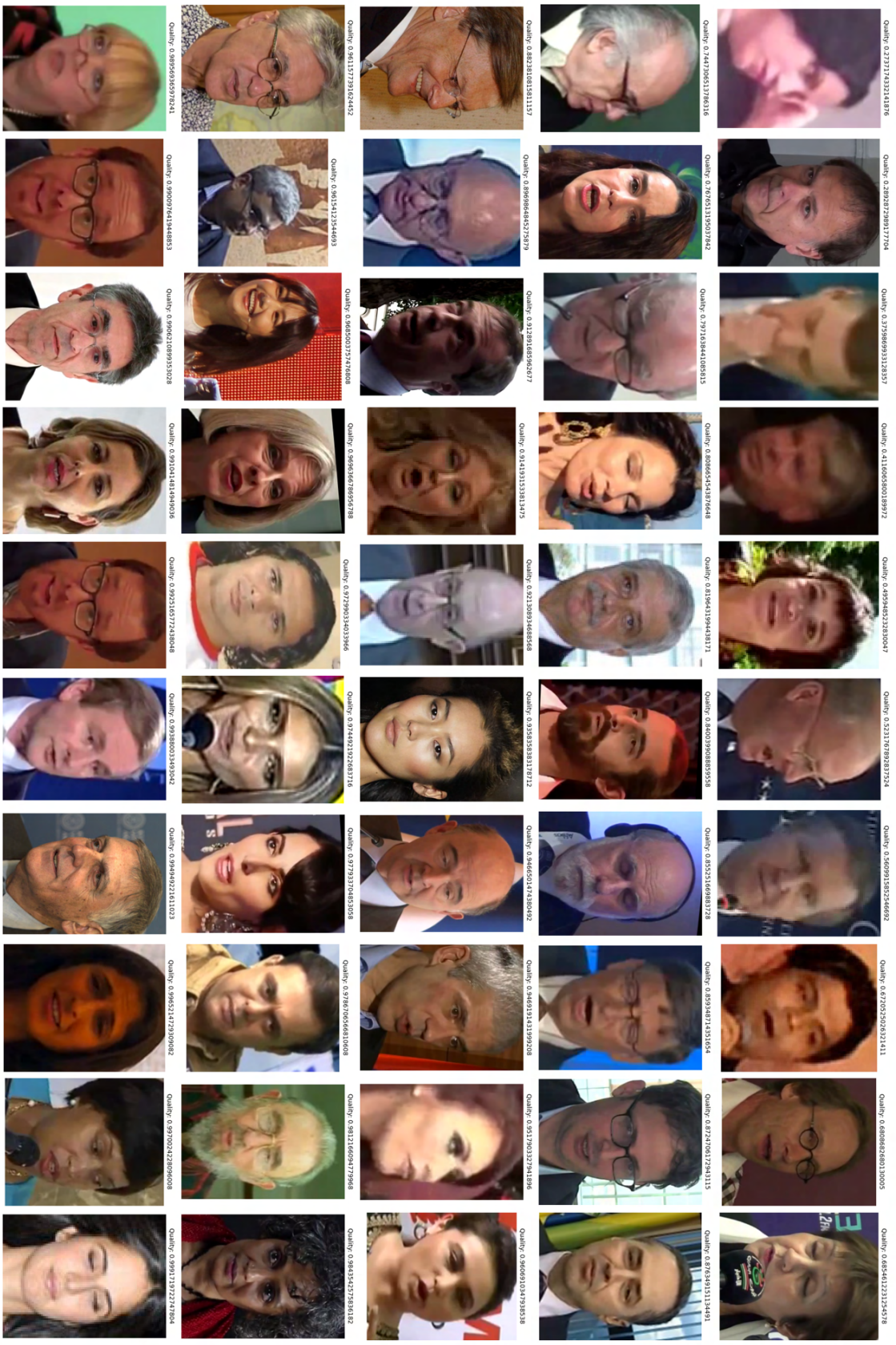


Figure B.11: Images sorted by quality for FaRL + MLP experiment.

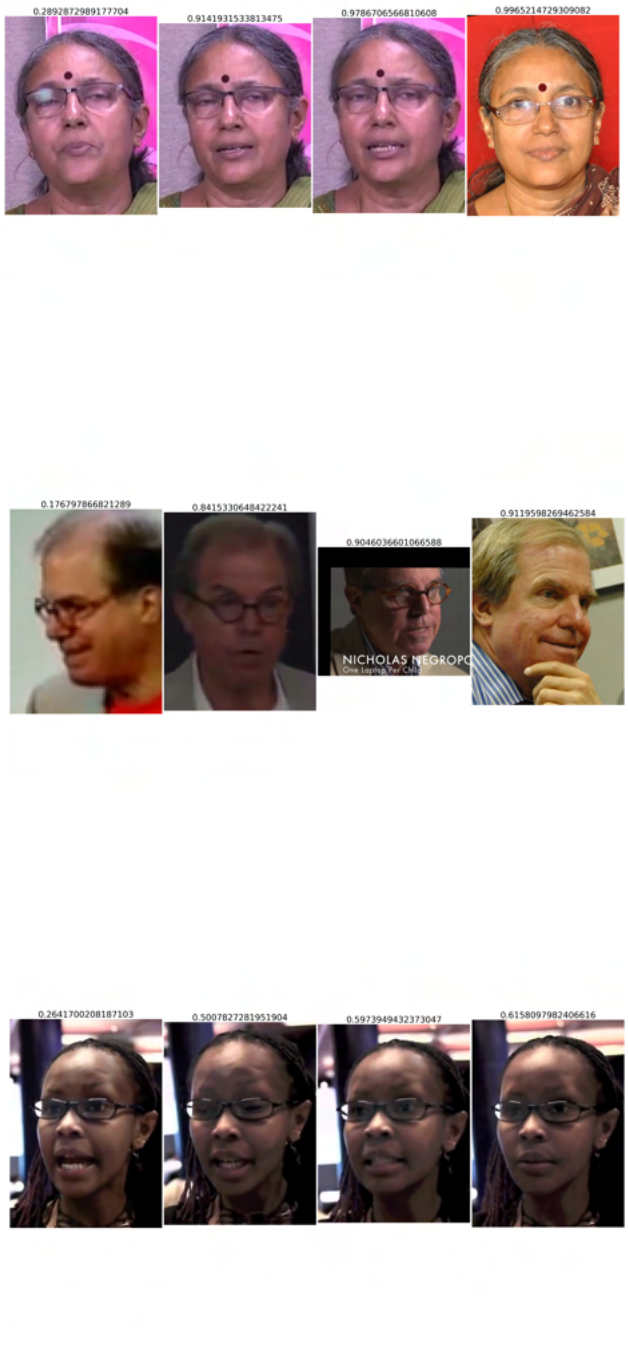


Figure B.12: Qualities for different persons in FaRL + MLP experiment.

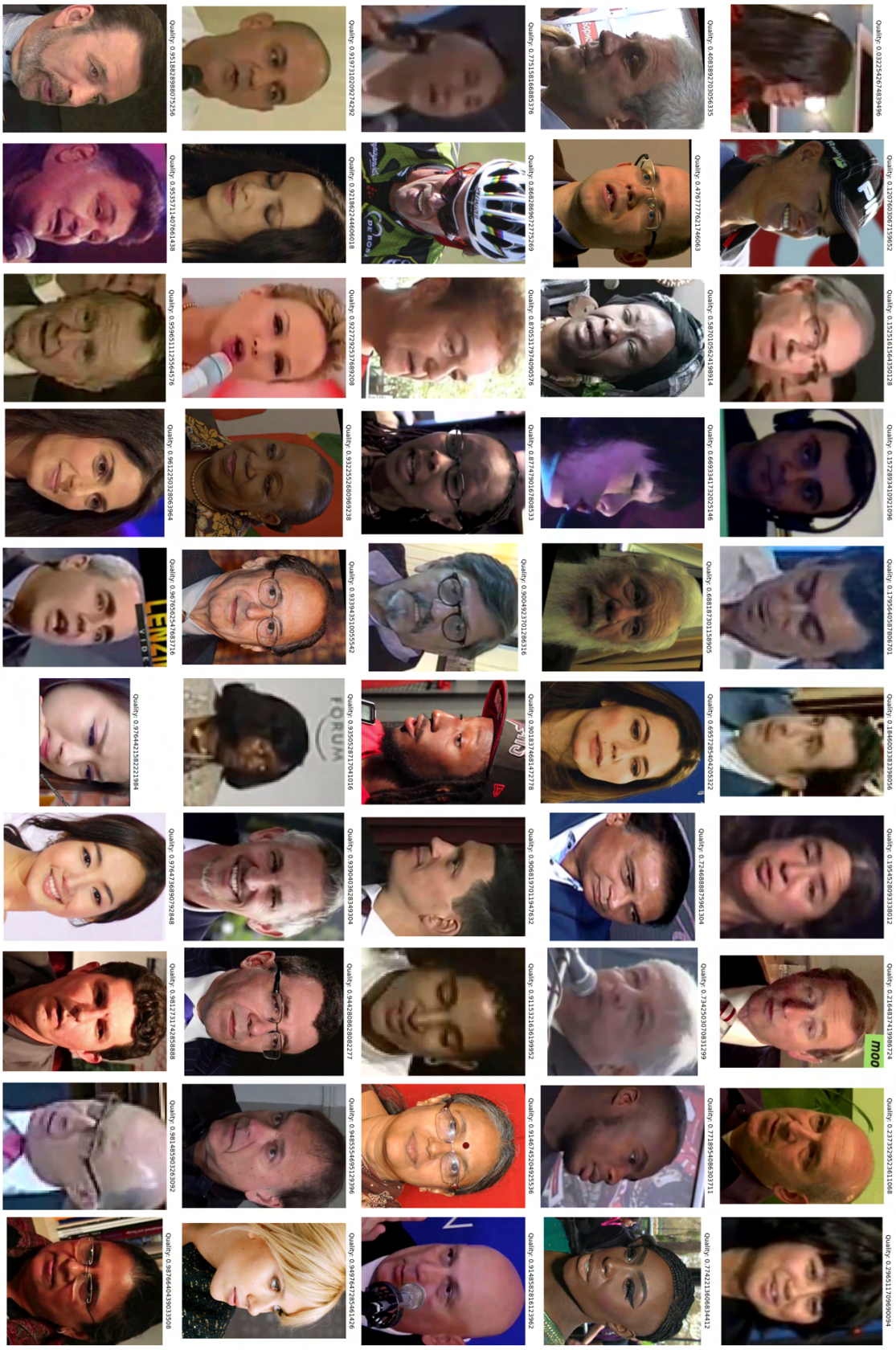


Figure B.13: Images sorted by quality for Bootstrapping + FaRL experiment.

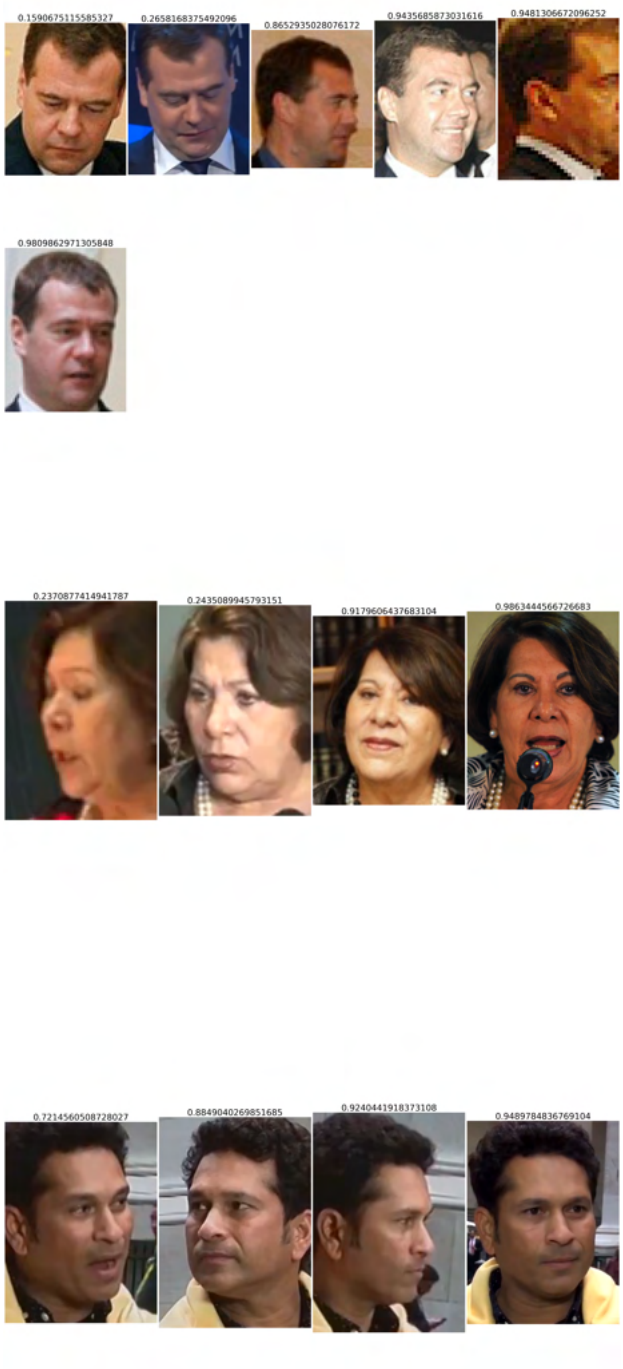


Figure B.14: Qualities for different persons in Bootstrapping + FaRL experiment.