

# Analysis of the Influence of Cosine Distance Threshold Values in a Real-time Face Recognition System

Maksym Holikov<sup>1</sup>, Volodymyr Donets<sup>1</sup>, Viktoriia Strilets<sup>1</sup> and Kyryl Korobchynskiy<sup>1</sup>

<sup>1</sup> Institute of Computer Science and Artificial Intelligence, V.N. Karazin Kharkiv National University, 4, Svobody, Sq., Kharkiv, 61022, Ukraine

## Abstract

The quality of a face recognition system largely depends on the correct choice of threshold value when comparing vector features (embeddings). This paper investigates the impact of cosine distance thresholds on the performance of a real-time face recognition system. The proposed approach combines face detection using MediaPipe FaceMesh and feature extraction using the ArcFace model. A series of experiments with different threshold values was conducted, the results of which were evaluated using the following metrics: Accuracy, Precision, Recall, False Accept Rate (FAR), and False Reject Rate (FRR). The results show that the choice of threshold directly determines the trade-off between security and convenience of the system. The optimal range of cosine distance threshold values was established as 0.05–0.07, which minimizes both FAR and FRR, which is important for practical use in video surveillance, access control, and user authentication systems.

## Keywords

Computer vision, deep learning, face recognition, cosine distance, real-time systems, MediaPipe, FaceMesh, ArcFace

## 1. Introduction

Biometric face recognition technologies are one of the most widespread and most researched areas in the field of computer vision. As a result of the development of deep neural networks, modern face recognition models have achieved a level of accuracy that is close to human perception on well-known test sets (LFW, MegaFace, IJB-C) [1]. Such systems are already widely used in video surveillance, access control, banking security, and mobile authentication.

A key step in the task of face recognition is calculating the distance between two embedding vectors representing faces in recognition models. Most often, cosine distance or its variations are used for this purpose [2]. However, the final decision on whether two samples match or differ depends on a threshold value that determines the balance between the False Accept Rate (FAR) and False Reject Rate (FRR) indicators [3]. A threshold value that is too low leads to frequent false rejections (failure to recognize even a real user), while a threshold value that is too high leads to false acceptances (identifying different people as the same person). Thus, the correct choice of threshold is crucial for the system's reliability.

The literature notes that optimal threshold values can vary significantly depending on the data set, shooting conditions, and even personal characteristics (race, age, gender) [4]. This creates a problem of threshold inconsistency, where the model demonstrates different effectiveness at the same threshold on different samples. For practical applications, a universal threshold is usually chosen, for example, one that provides an Equal Error Rate (EER) or a fixed FAR level, but accuracy may be reduced [5].

This work focused on an experimental study of the impact of cosine distance thresholds on face recognition quality. To this end, a pipeline was used that combines face detection with MediaPipe FaceMesh [6] and feature extraction using the ArcFace model [7]. A series of experiments with

<sup>1</sup> ProfIT AI'25: 5<sup>th</sup> International Workshop of IT-professionals on Artificial Intelligence, October 15–17, 2025, Liverpool, UK

✉ maksym.holikov@karazin.ua (M. Holikov); v.donets@karazin.ua (V. Donets); viktorija.strilets@karazin.ua (V. Strilets); k.korobchynskiy@karazin.ua (K. Korobchynskiy)

ORCID 0000-0003-4842-7823 (M. Holikov); 0000-0002-5963-9998 (V. Donets); 0000-0002-2475-1496 (V. Strilets); 0000-0002-3676-6070 (K. Korobchynskiy)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

different thresholds was conducted, and the results were evaluated using standard metrics (Accuracy, Precision, Recall, FAR, FRR) [5]. The results obtained allow determining the optimal range of thresholds for practical use and show the compromise between security and the convenience of the system.

## 2. Related work

In face recognition systems, two images are converted into embedding vectors (e.g., using FaceNet, ArcFace, CosFace, etc.), and the similarity between them is usually assessed using cosine similarity (or the corresponding cosine distance) [8]. A pair is considered a ‘match’ (the same identifier) if the similarity value exceeds a certain threshold; otherwise, it is considered a ‘non-match’. Classification errors are defined as False Accept (FA) – mistakenly accepting different individuals as one, and False Reject (FR) – mistakenly rejecting one individual as two. The corresponding FAR (False Accept Rate) and FRR (False Reject Rate) indicators depend on the selected threshold. The EER (Equal Error Rate) point corresponds to the threshold at which  $FAR = FRR$  [8]. The ROC (TPR–FPR) curve and the construction of DET nomograms are also obtained by searching through thresholds. Therefore, the choice of threshold value directly determines the quality indicators (Accuracy, Precision, Recall, etc.) [8].

Lowering the threshold (a more ‘lenient’ matching criterion) leads to an increase in FAR, when more impostors are mistakenly accepted, and a decrease in FRR, and vice versa when the threshold is raised. For example, when evaluating on LFW or other benchmarks, the standard technique is 10-fold cross-validation to select the optimal threshold, and most often it is the threshold value of cosine similarity [9]. However, many studies note that a threshold selected for one sample may not be suitable for another with a different origin, lighting, racial composition, etc. [9]. During training, the model can be optimised based on both the distance between feature vectors and the angle between them. In the case of the ArcFace and CosFace approaches, an additional angular or cosine shift is introduced, which increases the resolution of the feature space. However, at the verification stage, it is still necessary to determine the threshold value for making a decision. This threshold is often set based on the desired level of FAR or EER. The literature states that traditional approaches compare both classes (mated/impostor) with a fixed threshold, but ‘the best threshold for different classes is often different’ [9]. It is argued that the optimal threshold is usually specific to a particular dataset – the best thresholds for different datasets often differ. In practice, it is difficult to find the optimal threshold without access to test data [9].

Some works propose adaptive or coordinated threshold selection strategies that implement adaptive thresholding for each registered face in the database: instead of a single global threshold, they store a separate threshold for each sample, which leads to a significant improvement in accuracy (up to a 22% increase on LFW in their protocol) [9]. In [5], the authors highlight the discrepancy in thresholds across different domains and propose a new protocol called ‘One-Threshold-for-All’, which utilizes a single fixed threshold (referred to as the Calibration Threshold) for evaluating multiple datasets simultaneously. They show that the traditional approach of selecting a separate threshold for each dataset is inconsistent with the practical scenario of a single threshold and slows down the implementation of models. In [10], the concept of threshold inconsistency is introduced: even if the model is very accurate, different thresholds may be required for different classes to maintain the same FAR/FRR level, and the OPIS metric is proposed to measure the discrepancy of thresholds across classes, showing that solutions optimized for accuracy alone often worsen threshold consistency [10].

Other approaches focus on the selected operating mode. For example, in [11], the focus is on a fixed FAR level (‘Anchor FAR’) as a key criterion for practical FR systems: they optimize the goal of maximising TAR (True Accept Rate) at a given FAR, showing that different models are optimally different at different target FAR values [11]. Thus, the choice of threshold (and, accordingly, FAR) determines which model gives the best result.

A number of studies specifically investigate the role of cosine similarity in verification tasks. Work [12] analyses the distribution of cosine distances between positive and negative pairs in the complex DFW2019 dataset: it turns out that many ‘real’ pairs have low cosine similarity (due to face masking), which complicates threshold selection. This demonstrates that a fixed threshold on hard data can generate false rejections. Similarly, work [13] indicates that after training, the model has a fixed ‘cut-off level’ of cosine similarity, which is not entirely consistent with the testing procedure (where the threshold is strictly fixed). New loss functions also explicitly take the threshold into account: for example, [14] introduces USS Loss, which trains a single unified threshold for all pairs. Using 20 random identities as an example, it is shown that the optimal thresholds for them almost coincide with a single ‘unified’ threshold (about 0.4896). This indicates that this approach provides a more compact distribution of threshold values and simplifies the decision on a match.

### 3. Problem statement

The task of face verification is formulated as determining whether two images belong to the same person. To do this, each image is converted into a vector embedding using a neural network  $f(x) \in R^d$ , where  $d$  - the dimensionality of features (e.g., 512 in the case of ArcFace). For a pair of embeddings  $f(x_i), f(x_j)$  the cosine distance is calculated using the formula:

$$D(f(x_i), f(x_j)) = 1 - \frac{\sum_{i=1}^n (x_i y_i)}{\sqrt{\sum_{i=1}^n (x_i^2)} \cdot \sqrt{\sum_{i=1}^n (y_i^2)}}. \quad (1)$$

The system decides on a ‘match’ (1) or ‘no match’ (0) based on the threshold value  $\tau$ :

$$decision = \begin{cases} 1, & D(f(x_i), f(x_j)) < \tau \\ 0, & D(f(x_i), f(x_j)) \geq \tau \end{cases}. \quad (2)$$

The problem lies in choosing the optimal threshold value  $\tau$ . Depending on its value, the system indicators change:

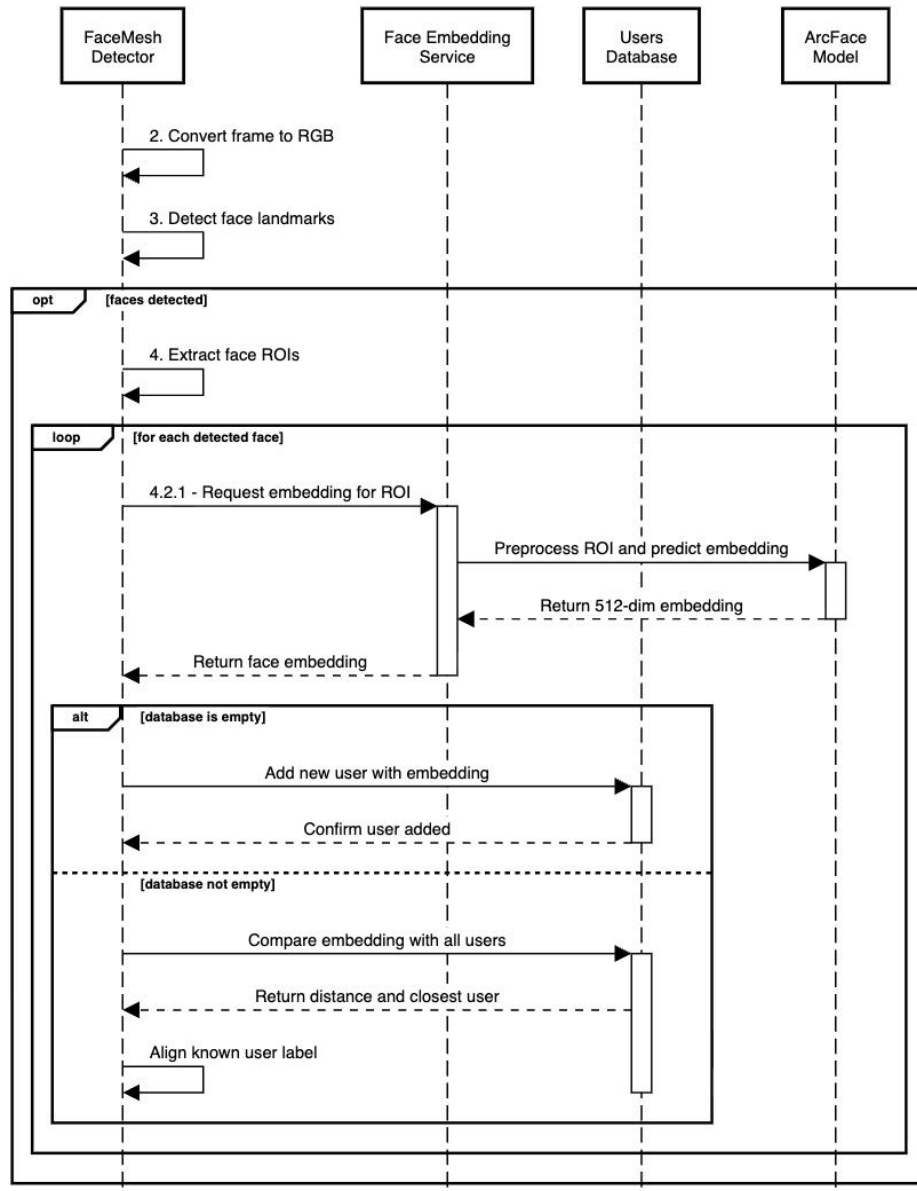
- False Accept Rate (FAR) – the proportion of cases in which different people are mistakenly identified as the same person;
- False Reject Rate (FRR) – the proportion of cases where a genuine user is rejected by the system;
- Accuracy, Precision, Recall – integral quality indicators;
- Equal Error Rate (EER) – the point where FAR = FRR, often used as a benchmark for selecting  $\tau$ .

In practical systems, it is impossible to find a universal threshold that will be equally effective for all scenarios and data sets. Too low a value reduces FAR but increases FRR, which degrades usability. Conversely, too high a value reduces FRR but increases FAR. Thus, the balance depends directly on the chosen threshold.

### 4. Face recognition system model

The input data for the face recognition system is a video stream from a camera. Each frame is sent to a pre-processing module, where image normalisation and basic transformations are performed to improve the stability of subsequent analysis stages. In particular, procedures for conversion to standard colour space, scaling, and noise filtering are applied.

The sequence diagram for face identification is shown in Figure 1:



**Figure 1:** Diagram showing the sequence of data flow and interaction between components in a facial recognition system.

As can be seen in Figure 1, the system's operation sequence unfolds in several main stages. Pre-processing of the frame is implemented in steps 2–3, where conversion to the standard colour space and key point detection are performed. Further face region selection (step 4) ensures the formation of an ROI for each face found. In step 4.2.1, the ROI is transferred to the vector representation formation service, where a 512-dimensional embedding is calculated using the ArcFace model. Next, depending on the state of the user base (whether the database is empty or the database is not empty branch), the embedding is either added as a new record or compared with existing vectors based on cosine distance. The final step is to return the closest user or add a new one, which is consistent with the subsequent text description.

Face detection in a frame is performed using key point analysis and facial geometry methods. The use of a topological grid allows not only to highlight the region of interest (ROI), but also to increase accuracy by taking into account variations in poses, expressions, and partial overlaps.

This method provides significantly greater accuracy and stability compared to classic trackers such as CSRT, MOSSE, or KCF. Traditional trackers focus on local patterns or pixel movement between frames and work well in relatively static conditions. However, they have a number of

limitations in dynamic environments: with partial overlaps, sudden changes in head position or facial expressions, trackers can lose the object, and the bounding box ‘slides’, leading to incorrect ROI formation and reduced accuracy of subsequent identification.

The use of a topological grid allows for the identification of over 400 key facial points, including the contours of the eyes, nose, mouth, and outer contour of the head. This ensures accurate ROI selection, which includes only the most relevant facial pixels, avoiding the background, hair, or other extraneous elements. High ROI detail directly affects the quality of the resulting vector representations (embeddings), increasing the accuracy of comparison and verification.

Another advantage of this approach is that classic trackers are often prone to ‘shifts’ when the user moves closer to or further away from the camera or when the lighting angle changes. Face Mesh, on the other hand, provides stable ROI detection regardless of such variations, and subsequent normalization of the region of interest ensures uniformity of vector representations for all frames.

In addition, this method of determining ROI does not depend on the initial frame or prior initialization. Unlike trackers, which lose the object when it disappears from the frame and require re-initialization, Face Mesh processes each frame independently. This makes the system more reliable in dynamic environments where users appear or disappear from the camera's field of view.

Thus, the use of a topological grid to determine ROI provides more accurate face detection, high resistance to changes in pose, lighting, and partial overlaps, as well as stability of the resulting embeddings. Compared to classical trackers, this approach increases the reliability of the face verification system and improves the quality of the final result, which is critical in real-time tasks and interactive user monitoring.

After ROI selection, the face image is converted into a compact vector representation — embedding. This is a multidimensional vector that encodes the most important features for identifying a person. The vector space is chosen so that the distances between points correspond to semantic proximity: two images of the same person are located close to each other, while images of different people are located at a relatively large distance.

In the proposed system, embeddings are used as a universal format that allows comparisons to be made regardless of lighting conditions, head position, or changes in appearance. This approach makes the method more generalised and less dependent on a specific data set, which is especially important for systems that need to work with new users without retraining.

Cosine distance is used to assess the degree of similarity between face embeddings. Cosine distance was chosen because it is invariant to the absolute length of feature vectors and can more accurately reflect the similarity between multidimensional representations of faces. Unlike Euclidean metrics, which can be sensitive to scale variations, cosine distance only evaluates the angle between vectors, making it more reliable in conditions of changing lighting or small variations in facial expressions.

Thanks to this approach, verification is reduced to the task of comparing numerical values and can be performed at high speed, which meets the requirements for real-time systems.

The final stage is the integration of verification results into the video stream. For each face detected in the frame, the system applies a corresponding label with the user ID or a ‘new’ mark.

The proposed approach has the following advantages:

- resistance to environmental dynamics;
- the algorithm operates in real time using vector distance calculations.

## 5. Results and discussion

The experiments were conducted on the LFW (Labelled Faces in the Wild) dataset [15], which is widely used to evaluate face verification algorithms. The dataset contains over 13,000 photographs of people taken in uncontrolled conditions, allowing for the simulation of real-life scenarios. The images show significant variations in lighting, head position, accessories (glasses, headwear), and

image quality, making LFW one of the most widely used standards for evaluating face recognition algorithms.

To construct test pairs, both positive examples (images of one person) and negative examples (different people) were selected.

The results for different threshold values of the cosine distance are given in Table 1.

**Table 1**

System performance indicators at different cosine distance threshold values

| Threshold | Accuracy | Precision | Recall | FAR | FRR    |
|-----------|----------|-----------|--------|-----|--------|
| 0.02      | 0.6579   | 1.0       | 0.6579 | 0.0 | 0.3421 |
| 0.03      | 0.8684   | 1.0       | 0.8684 | 0.0 | 0.1316 |
| 0.04      | 0.8947   | 1.0       | 0.8947 | 0.0 | 0.1053 |
| 0.05      | 0.9211   | 1.0       | 0.9211 | 0.0 | 0.0789 |
| 0.06      | 0.9737   | 1.0       | 0.9737 | 0.0 | 0.0263 |
| 0.07      | 1.000    | 1.0       | 1.000  | 0.0 | 0.0000 |
| 0.08      | 1.000    | 1.0       | 1.000  | 0.0 | 0.0000 |

As can be seen from the table, the choice of threshold value directly affects the accuracy of the system and the FAR/FRR error ratio.

Low threshold values (0.02 – 0.03) are too strict. At a threshold of 0.02, the FRR value reaches 34.2%, which means that one-third of genuine users fail the verification. Despite the absence of false acceptances (FAR = 0), this mode is unsuitable for real-world applications due to its low usability. Raising the threshold to 0.03 significantly reduces the FRR to 13.2% and increases accuracy to 86.8%, but the number of false rejections is still too high.

In the middle range (0.04–0.05), performance gradually stabilises. FRR decreases to 10.5% and 7.9% respectively, while accuracy increases to 92.1%. This already makes the system suitable for use in relatively controlled scenarios (e.g., office entrance with regular users). However, there is still a risk that some users will be falsely rejected.

At a threshold of 0.06, the system demonstrates very high performance: Accuracy = 97.4%, FRR = 2.6%, FAR = 0. This means that only 1 in 38 genuine users may be rejected, with no false acceptances recorded. This result is the most balanced and practically significant: the system becomes user-friendly while maintaining a high level of security.

Starting from a threshold of 0.07, the system achieves perfect results — Accuracy, Precision, and Recall are 100%, and FAR and FRR are zero. From a technical point of view, this means that no errors were recorded in the test sample. However, as previous research shows, achieving ‘perfect’ results is often explained by the limited or homogeneous nature of the sample. In real-life scenarios — with different lighting, poses, accessories (glasses, masks) — it is practically impossible to avoid errors. Therefore, such a result should be considered more as an artefact of a specific experiment rather than a guarantee of absolute reliability.

However, the obtained ‘ideal zone’ (0.07–0.08) can be partially explained by the specifics of the test data set, since in practice there is always noise, variations in lighting, poses, and appearance, which make it impossible to achieve absolute indicators. This is consistent with the well-known

problem of threshold inconsistency, where the optimal threshold depends on the conditions of use and sample characteristics.

Thus, the optimal operating range for the system can be defined as 0.05–0.07, where the best compromise between minimizing FRR and maintaining zero FAR is achieved. This result is important for practical real-time applications such as video surveillance, where even a single impostor acceptance error can have critical consequences.

## 6. Conclusions

This paper investigated the impact of cosine distance threshold values on the performance of real-time face recognition systems. Experimental results showed that the choice of threshold directly determines the balance between FAR and FRR metrics, as well as the overall accuracy of the system.

It was established that:

- thresholds that are too low (0.02–0.03) result in high FRR, which reduces usability;
- in the range of 0.05–0.06, the system demonstrates an optimal compromise between security and accessibility, ensuring high Accuracy values and zero FAR;
- starting from a threshold of 0.07, the test set shows perfect results (100% Accuracy, Precision, and Recall), but this result is likely due to the characteristics of the sample and requires additional verification on more heterogeneous data.

Thus, the optimal operating threshold for the system under study can be determined as 0.05–0.07, which minimises the number of false rejections without the risk of accepting an imposter. The conclusions obtained are important for the practical implementation of face verification technologies in video surveillance, access control, and user authentication tasks.

Further research could focus on testing the stability of the optimal threshold on different data sets, developing adaptive thresholding methods for specific users, and integrating additional factors (lighting conditions, dynamic scenes, changes in appearance) that affect the accuracy of the system.

## References

- [1] Y. Wu, “Rotation consistent margin loss for efficient low-bit face recognition,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6866–6876.
- [2] Datacamp, “What is Cosine Distance?”, 2024. URL: <https://www.datacamp.com/tutorial/cosine-distance>.
- [3] Zaidi, A.Z., Chong, C.Y., Jin, Z., Parthiban, R., and Sadiq, A.S., Touch-based continuous mobile device authentication: State-of-the-art, challenges and opportunities, J. Network Comput. Appl., 2021, vol. 191, p. 103162. <https://doi.org/10.1016/j.jnca.2021.103162>.
- [4] Cavazos, J. G., Phillips, P. J., Castillo, C. D., and O’Toole, A. J. (2019). Accuracy comparison across face recognition algorithms: where are we on measuring race bias? arXiv preprint arXiv:1912.07398. doi: 10.48550/arXiv.1912.07398.
- [5] Jiaheng Liu, Zhipeng Yu, Haoyu Qin, Yichao Wu, Ding Liang, Gangming Zhao, and Ke Xu. 2022. OneFace: One Threshold for All. In Eur. Conf. Comput. Vis. Springer.
- [6] GitHub, Google AI Edge MediaPipe Face Mesh, 2020. URL: <https://github.com/google-ai-edge/mediapipe/wiki/MediaPipe-Face-Mesh>.
- [7] Huang, Y. H., & Chen, H. H. (2022). Deep face recognition for dim images. Pattern Recognition, 126, 108580.
- [8] S. Kilany and A. Mahfouz, “A comprehensive survey of deep face verification systems adversarial attacks and defense strategies”, Scientific Reports, vol. 15, no. 1, Aug. 2025. doi: 10.1038/s41598-025-15753-8.

- [9] H.-R. Chou, J.-H. Lee, Y.-M. Chan, and C.-S. Chen, “Data-specific adaptive threshold for face recognition and authentication”, 2019 IEEE Conference on Multimedia InformationProcessing and Retrieval (MIPR), Mar 2019. doi: 10.1109/MIPR.2019.00034.
- [10] Q. Zhang, L. Xu, Q. Tang, J. Fang, Y. N. Wu, J. Tighe, and Y. Xing, “Threshold-consistent margin loss for open-world deep metric learning”, arXiv, 2023. doi: 10.48550/arXiv.2307.04047.
- [11] Liu, J.; Qin, H.; Wu, Y.; Liang, D. Anchorface: Boosting tar@far for practical face recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 22 February–1 March 2022.
- [12] Deng, J., Zafeririu, S.: ArcFace for disguised face recognition. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 485–493. IEEE, Piscataway (2019).
- [13] Liu, L.; Chen, M.; Chen, X.; Zhu, S.; Tan, P. GB-CosFace: Rethinking softmax-based face recognition from the perspective of openset classification. arXiv 2021, arXiv:2111.11186.
- [14] Q. Li, X. Jia, J. Zhou, L. Shen, and J. Duan, “UniTSFace: Unified threshold integrated sample-to-sample loss for face recognition”, arXiv, 2023. doi: 10.48550/arXiv.2311.02523.
- [15] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. University of Massachusetts, Amherst, Technical Report 07-49, 2007.