# Cyshield AI Assessment - CVCY001 Task

Ahmed Tamer Samir

Submitted to Cyshield Computer Vision Team

December 11, 2025

# Contents



# List of Figures

# 1 System Architecture

The proposed age-invariant face-matching system consists of four main modules (see Figure 1 above). Each input image of the pair follows the same path until the final decision stage.

1. **Preprocessing & Face Alignment Pipeline**
   Raw images are processed by MTCNN to detect faces and extract five facial landmarks. An affine transformation aligns the face (eyes on the same horizontal line, normalized inter-ocular distance), followed by tight cropping and resizing (160×160 for the embedding model, 224×224 for the age model).

2. **Age Prediction Branch**
   A ResNet-50 (pretrained on ImageNet and fine-tuned for age regression) takes the aligned face and outputs a continuous apparent age estimate, which is later used as auxiliary information.

3. **Face Embedding Extraction**
   The aligned face is fed to a frozen InceptionResNet-v1 (weights = 'vggface2' from facenet-pytorch), producing a 512-dimensional identity embedding.

4. **Final Matching Decision (MLP Head)**
   Features fed to the MLP are: • Absolute difference of embeddings $|\mathbf{e}_1 - \mathbf{e}_2|$ • Absolute age difference $|a_1 - a_2|$ • Individual ages $[a_1, a_2]$

This design significantly improves performance on large age gaps compared to using only cosine similarity on raw embeddings.

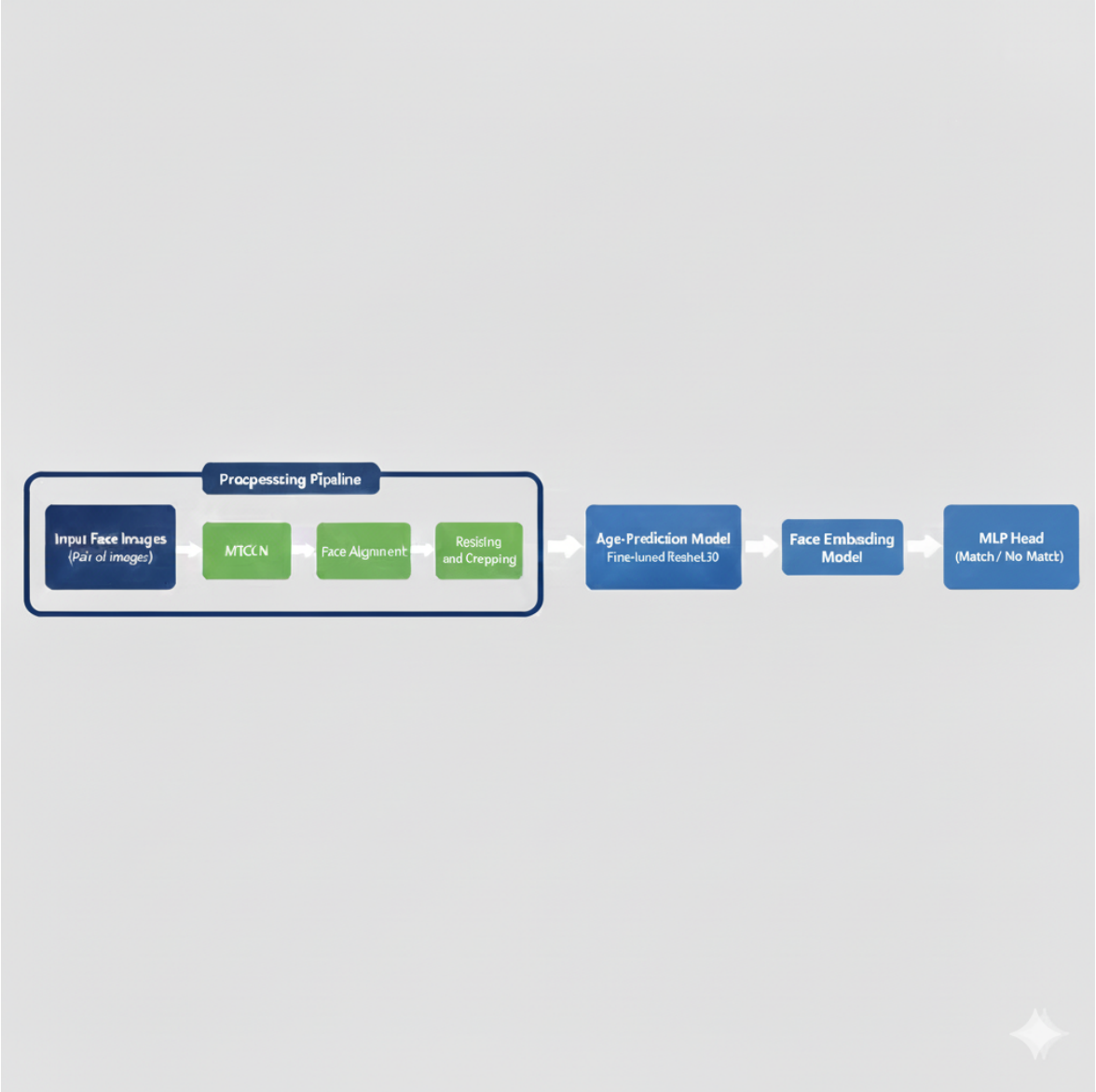


Figure 1: High-level pipeline of the proposed age-invariant face-matching system. The same preprocessing → age prediction → embedding steps are applied independently to both images before the MLP head produces the final Match / Non-Match decision.

# 2 Dataset Selection and Rationale

The system employs two fine-tuned models, each trained on carefully collected datasets.

## 2.1 Age Prediction Model Dataset

For the age prediction model, we constructed a diverse composite dataset by merging three publicly available sources to address specific limitations and enhance generalization.

- **Base dataset**: UTKFace (∼20,000 images). Although large and well-labeled, it has two major shortcomings:
  1. Severe under-representation of children aged 0–13.

2. Insufficient samples of older adults (age 40+).

- To address the lack of young samples and add longitudinal information, we incorporated **FG-NET** (∼1,000 images). This dataset contains multiple images of the same individuals taken at different ages, which is especially useful for cases where facial appearance changes only minimally over certain age ranges (e.g., from 17 to 25 years). Including these sequences helps the model better handle "slow-aging" individuals.

- To improve coverage of older age groups, we added the cropped face version of the **APPa-Real** dataset from Kaggle (∼7,950 images), which provides a substantial number of samples for individuals over 40 years old.

A further benefit of combining datasets from multiple sources is the increased diversity in image quality, resolution, lighting conditions, and face poses, resulting in a more robust and realistic training distribution.

An important note : I truncated ages greater than 60, because of they have very low number of samples over the internet, so the model data is between age 0 and 60

## 2.2 MLP Dataset

### 2.2.1 Data Collection

The dataset comprises image pairs classified into two categories: positive pairs (two images of the same individual) and negative pairs (two images of different individuals). This binary classification dataset was constructed to train an MLP-based verification head, whose necessity is justified in Section 3.3.

- **Dataset Composition:** A total of 16,132 face pairs were collected, consisting of both positive samples (label = 1) and negative samples (label = 0).

- **Positive Sample Sources:** Positive pairs were sourced from two aging-specific datasets: FG-NET and MORPH. These pairs were selected with the constraint that the age difference between the two images must fall into one of three categories: 5+ years, 10+ years, or 20+ years.

- **Negative Sample Sources:** Negative pairs were extracted from the UTKFace dataset.

- **Dataset Selection Rationale:**

  1. FG-NET and MORPH were chosen for positive samples because they provide longitudinal facial images of the same individuals captured at different ages, making them ideal for age-invariant face verification.
  2. UTKFace was selected for negative samples due to its diverse collection of facial images from distinct individuals across various demographics.

### 2.2.2 Feature Engineering Pipeline

Following data collection, all image pairs underwent a systematic preprocessing and feature extraction pipeline to construct the final 1539-dimensional input vector for the MLP classifier.

**Preprocessing Stage:**

- Face detection and alignment were performed using the Multi-task Cascaded Convolutional Network (MTCNN), which localizes facial landmarks and normalizes face orientation and scale.

**Feature Extraction Stage:**

- **Age Estimation:** A ResNet50-based regression model predicted the ages of both individuals: $\text{age}_1$ and $\text{age}_2$.

- **Face Embedding:** FaceNet (InceptionResnetV1 architecture pre-trained on VGGFace2) extracted 512-dimensional embedding vectors for each face: $\mathbf{e}_1, \mathbf{e}_2 \in R^{512}$.

**Final Input Vector Construction:** The extracted features were concatenated to form a comprehensive 1539-dimensional representation:

$$\mathbf{x} = [\text{age}_1, \text{age}_2, \mathbf{e}_1, \mathbf{e}_2, |\text{age}_1 - \text{age}_2|, |\mathbf{e}_1 - \mathbf{e}_2|] \tag{1}$$

This vector serves as the input to the MLP classifier for binary verification.

# 3 Models Development

## 3.1 Age Prediction Model

### 3.1.1 Classification vs Regression Choice

For this task, I formulated age prediction as a regression problem rather than a classification problem for several reasons:
1. **Higher precision**: For instance, if a person is 23.5 years old, a regression model can predict 23.5, whereas a classification model would assign it to a discrete bin (e.g., 20–25).
2. **Avoiding hard boundaries**: In classification, two individuals aged 24.9 and 25.1 would fall into different classes, which is undesirable.
3. **Industry standard**: Most literature treats age prediction as a regression task rather than classification.

### 3.1.2 Model Architecture

I compared EfficientNet-B1 and ResNet50 , but at the end I decided to use ResNet50:

| **Feature** | **EfficientNet-B1** | **ResNet50** |
|---|---|---|
| Memory Usage | Lower | Higher |
| Top-1 Validation Accuracy | 79.1% | 76% |
| Number of Parameters | Fewer | More |
| Inference Time on Hardware (CPU/GPU) | Slower | Faster |

Table 1: Comparison between EfficientNet-B1 and ResNet50

**Reason for choosing ResNet50:**
Since memory was not my major concern (training was done on Colab Pro),also Inference Time on Hardware isn't my concern as I trained on Colab, also there isn't huge difference in there Validation accuracies, so I selected ResNet50 pretrained on ImageNet due to its higher number of parameters, which allows the model to capture more complex patterns for the regression task.

### 3.1.3 Fine-tuning Strategy

A progressive fine-tuning approach was applied in two phases:

1. **Phase 1:** Freeze the ResNet50 backbone and train only the custom regression head for 10 epochs.
2. **Phase 2:** Keep the regression head trainable, unfreeze the top two convolutional blocks (blocks 4 and 5) of ResNet50, and continue training for 50 epochs with a reduced learning rate. This helps preserve the pretrained weights of these layers while allowing them to adapt to the regression task.

## 3.2 Face-Matching Model

### 3.2.1 Model Selection

In chose a pretrained model (InceptionResnet-v1) which is pretrained on vggface2 dataset that contain 3.31 million images of 9131 identities, because it produces high-dimensional embeddings (512-D) that capture intrinsic identity features while being robust to variations in age, pose, and expression. Its training on a large and diverse dataset enables generalization across different age groups, which is essential for matching faces of the same person at different ages.

### 3.2.2 Limitations of the Face Matching Model

The face matching model, when paired with cosine similarity, has several limitations, especially when dealing with variations in age or physical appearance. These shortcomings stem from the following factors:

1. When comparing images of the same person taken many years apart (typically 10 or more years), or featuring significant appearance differences—such as a bearded versus clean-shaven look—the model may fail to identify them as the same individual, even though the identity is clearly the same (see Figure 2).


Figure 2: Example of appearance variation: Mo Salah with beard vs. without beard using Face matching Model then cosine similarity

2. Changes in hairstyle or hair color can substantially alter the embedding vectors. For instance, images of a woman with her natural hair color versus dyed hair might be mistakenly treated as belonging to two different people.

3. The use of a fixed cosine similarity threshold of 0.5 can lead to errors in borderline cases. When the similarity score is very close to the threshold (e.g., 0.48), the system may wrongly classify the faces as belonging to different individuals. This problem was observed with images of myself when i was young and i was old , which were incorrectly classified by Face-matching model and cosine similarity (see Figure 3).


Figure 3: Example of appearance variation due to age in Ahmed images using Face matching Model then cosine similarity

To address this issue, two possible solutions were considered:

1. Fine-tuning the face embedding model on a large dataset (a more difficult approach).
2. Adding a multilayer perceptron (MLP) head to make the final binary decision (an easier and more practical approach).

Given data-related constraints, the MLP head was chosen. The implementation details of this MLP head will be discussed

## 3.3 MLP Head

MLP Head was used in the system for the final decision if 2 faces match or not.

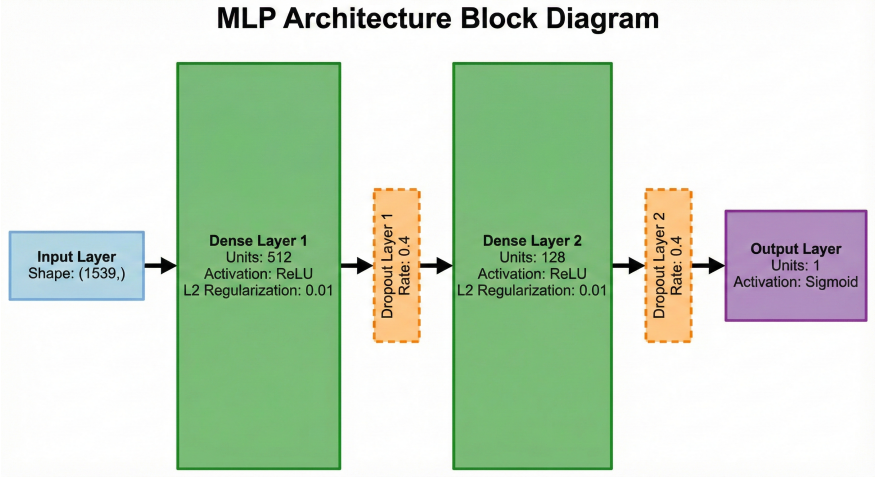


Figure 4: Architecture of the MLP Head used for face verification.

### 3.3.1 Fine-tuning strategy

ALL MLP layers were fine-tuned for 20 epochs with batch size=64 , and early stopping and Learning Rate Scheduler to avoid MLP overfitting to dataset

### 3.3.2 MLP Positive Effect

As we said before, the MLP was used to solve the limitations of the Face Matching Model and cosine similarity for the final decision. The MLP was trained on pairs of images of the same person across wide age ranges and with different facial appearances (beard vs. no beard, glasses vs. no glasses) to enable it to learn more complex facial patterns.

The following two examples, which previously failed with the Face Matching Model combined with cosine similarity, are now correctly classified after replacing cosine similarity with the MLP head, demonstrating its effectiveness (see Figures 5 and 6).


Figure 5: Mo Salah with beard vs. without beard: correctly classified as the same person using Face Matching Model + MLP head (previously failed with cosine similarity)

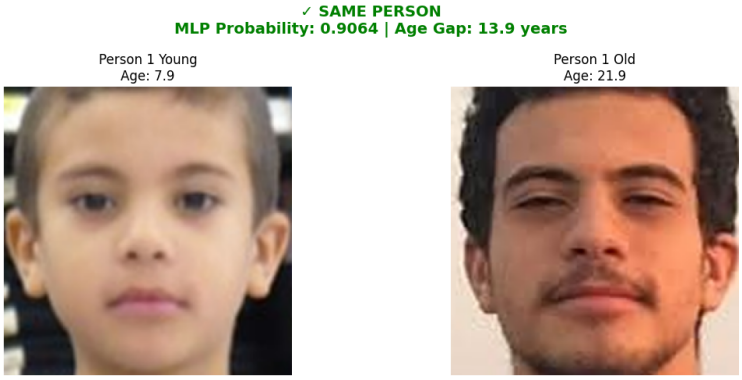


Figure 6: Ahmed young vs. old: correctly classified as the same person using Face Matching Model + MLP head (previously failed with cosine similarity)

# 4 Loss Function Selection and Reasoning

## 4.1 Age prediction Model Loss function

MSE is used because age prediction is a continuous regression task, and MSE penalizes larger prediction errors more strongly, guiding the model to predict ages accurately.

## 4.2 MLP Loss function

The Binary cross entropy loss function was chosen because the model performs binary classification, and this loss effectively measures the difference between the true labels (0 or 1) and the predicted probabilities from the sigmoid output, guiding the model to correctly predict the correct class

# 5 Performance Analysis and Evaluation Metrics

## 5.1 Age Prediction Model Metric

MAE was used a metric for this regression task

## 5.2 MLP Metric

Accuracy was used as a metric for MLP Performance , as we already achieved data balance when training this MLP

Table 2: Final Accuracies of the MLP Model

| **Dataset** | **Accuracy** |
|---|---|
| Training | 0.9770 |
| Validation | 0.9664 |
| Test | 0.9831 |

## 5.3 System Integrated Performance Analysis

Also after computing metrics , i created a folder containing multiple image samples for me when i am young and old , and also for known celebrities (because their face progression across ages could be a good way to test this system and most of us knows the expected output)