CYSHIELD

Face Matching Across Ages System

Technical Report

Computer Vision Assignment

Task: CYCV001

Submitted to

Cyshield – Computer Vision Team (Hiring Process)

Submitted by

Ahmed Gamal Noureddine

Email: eng.ahmed.gamal.cu@gmail.com

Date: 3 September 2025

Contents

1. Objective

This system determines whether two input photos belong to the same individual at different ages. The pipeline detects and crops faces (using MTCNN), predicts each face's age (using custom ResNet-50 age regressor trained on UTKFace dataset), produces 512-D facial embeddings (using InceptionResnetV1/FaceNet pretrained on VGGFace2), and decides "MATCH / NOT MATCH" using cosine similarity with a certain threshold. The design balances accuracy, speed, and implementation simplicity while remaining extensible for future improvements.

2. System Architecture & End-to-End Flow

2.1 High-Level Modules

- Input Handler: accepts exactly two input images (JPG/PNG).
- Face Detection & Cropping: MTCNN finds the largest face per image and returns a tight crop.
- Age Prediction: custom from scratch ResNet-50 regresses age (in years) from the cropped face (trained on UTKFace).
- Face Embedding: InceptionResnetV1 (FaceNet family, pretrained on VGGFace2) produces a 512-D feature vector.
- Similarity & Decision: L2-normalize embeddings, compute cosine similarity; compare with a threshold to output MATCH/NOT MATCH.
- Reporting: prints age1, age2, cosine similarity (as %), and the final decision.

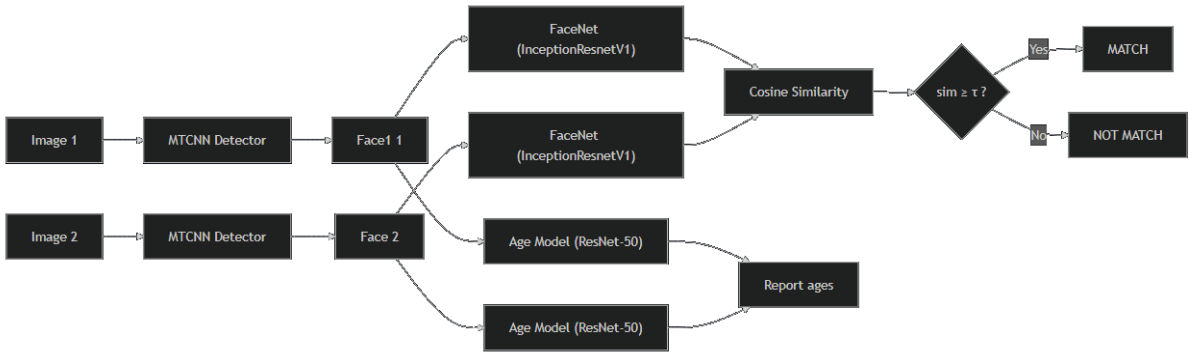

Figure 1. Full pipeline of the system

2.2 Data Flow (Input → Output)

1. Read and preprocess two input images.

Figure 2. Two sample input for the same person in different ages

2. For each image, run MTCNN to get squares cropped face (largest face with a margin).

Figure 3. The two detected faces by MTCNN

3. Age model: preprocess the 224×224 faces (with ImageNet mean/std) using custom from scratch ResNet-50 → scalar age in years.

4. Embedding model: preprocess to 160×160 → 512-D embedding → L2 normalization.
5. Cosine similarity: sim = e1 · e2
6. Decision: if sim ≥ τ (default 0.50), output MATCH; otherwise NOT MATCH then display ages, similarity, decision.

```
First person is: 14.91 years old

Second person is: 19.78 years old

Similarity: 65.51%

Faces MATCH ... This is the same person
```

Figure 4. System terminal output

2.3 Key Implementation Notes

- Detector: MTCNN(image_size=224, margin=40, keep_all=False, select_largest=True), this ensures that the model return one face per image with a margin too make the cropped face fit the age prediction model
- Embedding: InceptionResnetV1 pretrained on VGGFace2, it outputs 512-D vector.
- Threshold: default $\tau = 0.50$ tuned on test pairs. It balances the need for large threshold for small ages (children are confusing in matching) and the need for small threshold for accurately matching high age differences.
- We can also tune the threshold based on the application for which we use face matching system. It will depend on our interest in precision or recall, if we are interested in high precision we may increase the threshold, but for high recall we need to decrease the threshold.

3. Dataset Selection & Preprocessing

3.1 Dataset: UTKFace

I used UTKFace dataset which has wide age coverage (children to elderly) and pruned max 60 years old to speed up training as I trained from scratch. It has variable pose, illumination, and background. But UTKFace dataset can contain label noise and demographic imbalance.

3.2 Splits and Sampling

- Split: 80% training (18965 samples) and 20% validation (4742 samples).
- Sampling: Pruned the images of age bigger than 60 years to speed up training and remove extreme outliers.

3.3 Preprocessing and Augmentation

- Training Data Augmentation:
  - Random Resized Cropping of size = 224 and scale = 0.9~1.0
  - Random Horizontal Flip with probability = 0.5
  - Color Jitter with brightness = 0.2, contrast = 0.2, and saturation = 0.2
  - Random Grayscale with probability = 0.1
  - Gaussian Blur with kernel size = 3 and sigma = (0.1, 1.5)
  - Normalization with ImageNet mean/std.
- Validation Data Preprocessing:
  - Normalization with ImageNet mean/std.

4. Age Prediction Model Architecture (From Scratch ResNet-50)

4.1 ResNet-50 Architecture

- Input: 3×224×224 cropped face.
- Stem: 7×7 conv → batch norm → ReLU → maxpool.

- Residual stages: {3, 4, 6, 3} bottleneck blocks (1×1 → 3×3 → 1×1) with identity/projection shortcuts.
- Head: Global Average Pool → Linear(2048 → 1) → scalar age.
- Stabilization: zero-init the final BN scale in bottleneck branch to ease optimization. This reduce the effect of residual block at first of training and they found this better.
- Output: scalar age in years (clamped to 60 max).
- This model has 23,510,081 parameters (backbone + linear head).

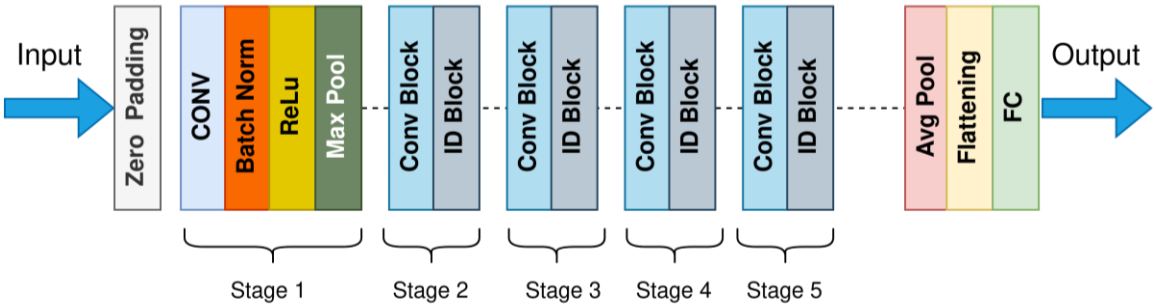

Figure 5. ResNet-50 model architecture

4.2 Loss Function

- First, I used mean absolute error (MAE) loss because it expresses the age difference well (MAE = actual age error)
- But I found that the model age error (actual age – predicted age) is very high for the small ages. The MAE was higher than 8 years.
- So, I then used Smooth L1 (Huber) Loss with beta = 6 for quadratically penalizing age errors less than 6 years to be robust to noisy labels.
- The Smooth L1 (Huber) performed better and ended with better age error.

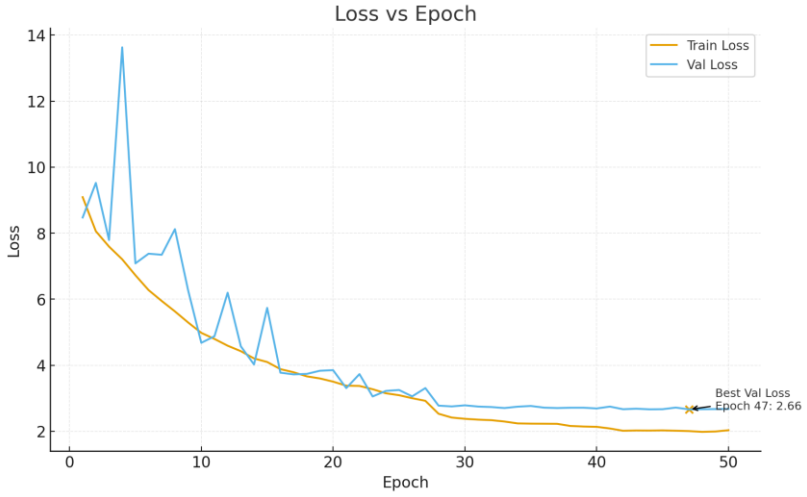

Figure 6. Smooth L1 (Huber) Loss

4.3 Training Setup

- Optimizer: SGD with learning rate = 0.005; momentum = 0.9, and weight decay = 0.0001 for regularization.
- Learning Rate Schedule: Reduce LR On Plateau with factor = 0.1 and patience = 3 epochs.
- Batch Size: 64 samples/batch
- Number of Epochs: 50 Epochs.
- Stability: gradient clipping (5.0) to avoid spikes.

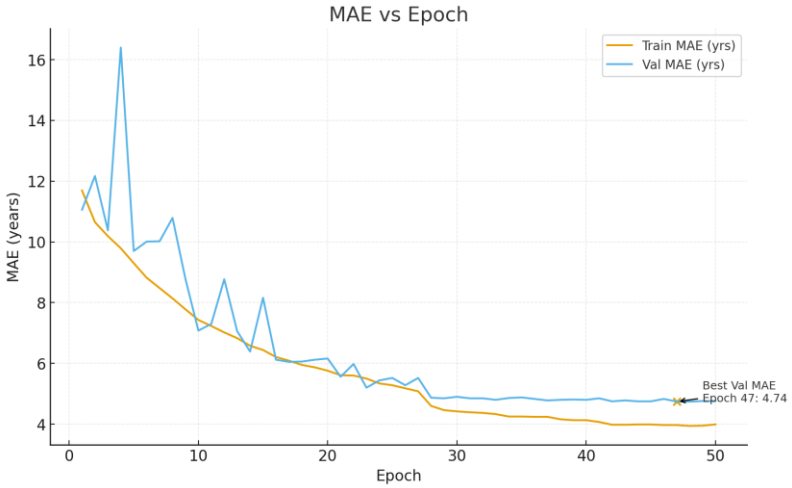

Figure 7. Age mean absolute error in years

5. Face Detection and Embedding Models

- Face Detection (MTCNN): keeps largest face with a margin (40 pixels) to include contextual pixels. I found this better for the age predictor model.

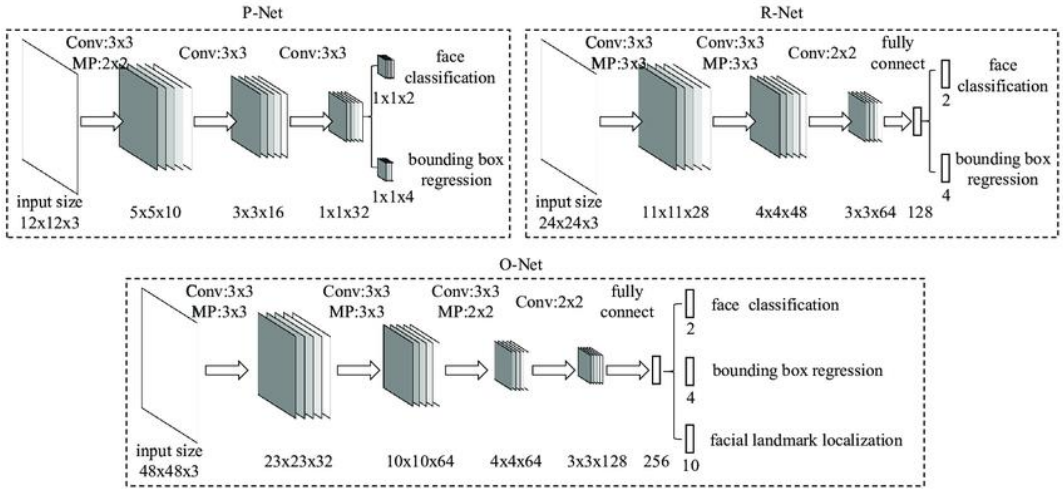

Figure 8. MTCNN model architecture

- Face Embedding (InceptionResnetV1/FaceNet): input face of dimensions 160×160 normalized to [−1,1]. It generates 512-D L2-normalized embedding.

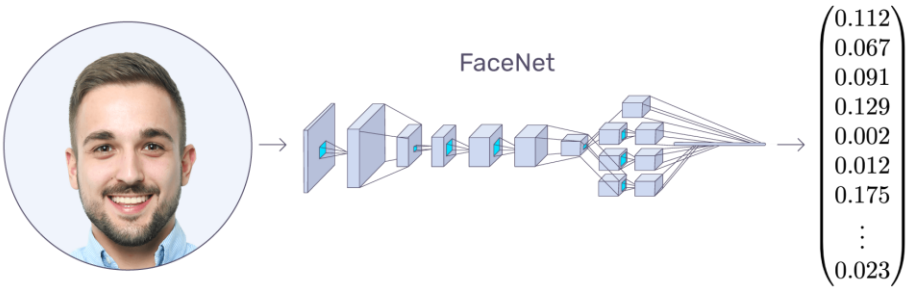

Figure 9. FaceNet embedding model

- Similarity & Decision: cosine similarity on embeddings; MATCH if sim ≥ threshold.

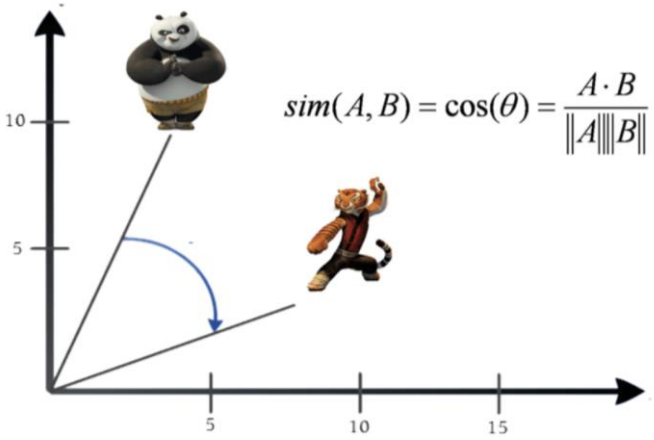

Figure 10. Cosine Similarity

6. Evaluation Metrics of Age Regressor

- MAE (Mean Absolute Error, in years): 4.76 years
- Calibration Plot: true ages (with 5-year bins) vs mean predicted vs true.

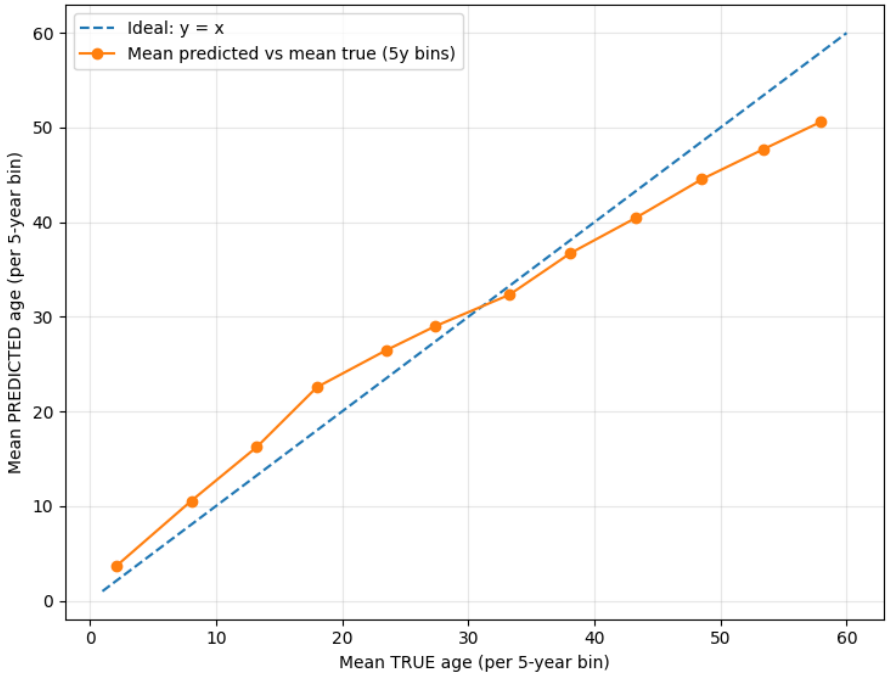

Figure 11. Mean predicted vs mean true values

- MAE by Age Group: 0–10, 11–20, 21–30, 31–40, 41–50, 51–60

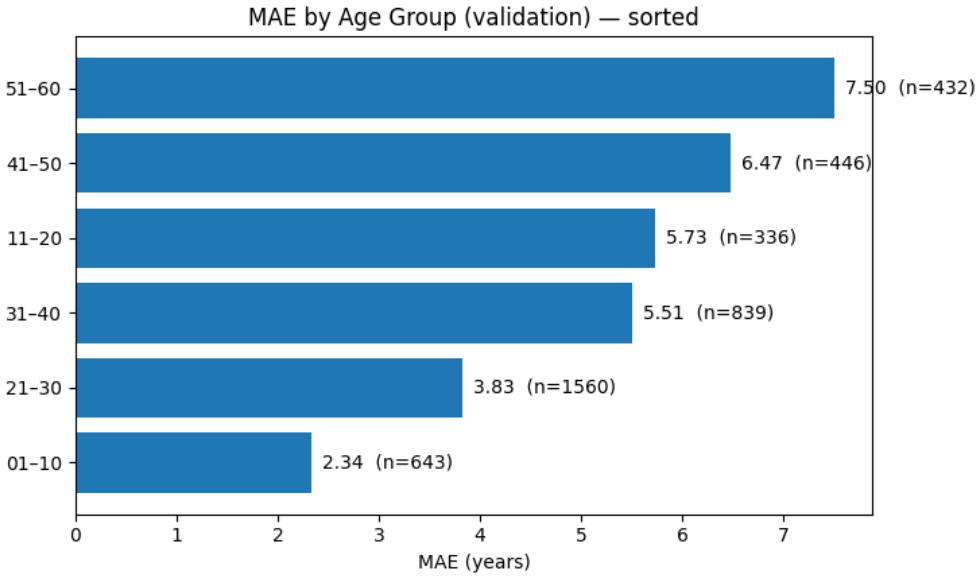

Figure 12. Mean absolute error for six age groups

7. Strengths & Limitations

7.1 Strengths

- Robust in face matching of the same person with moderate age differences, especially for clear, frontal faces and good lighting.
- Age predictor performs well on children and younger adults with strong MAE for ages 1–10 years and generally decent performance for under 40 years ages.
- Strong pretrained MTCNN face detector and FaceNet embedding model.
- Fast, modular pipeline; easy to deploy and extend.

7.2 Limitations

- Face matching would not perform well on images of lookalike people like brothers and twins.
- Age regression errors are larger for ages bigger than 40 years, as in these ages people does not change a lot, it is even hard for humans to predict their ages well.
- Age predictor is biased to this dataset faces. It cannot generalize well anywhere and MAE may drop on other demographics, sensors, and image styles.
- Global threshold may not generalize across domains, it would be better to calibrate it using domain test data.