PROJECT ABSTRACT

Face Age Estimation using CNN

A ResNet-18 CNN that estimates age from face photos, trained on UTKFace, delivered as a built-to-order student project with code, trained weights, report, PPT and viva kit.

01. Title

Face Age Estimation using CNN

02. Introduction / Background

Humans are unreliable at guessing age from a photograph, yet the cues are present — skin texture, facial proportions, hairline. Deep learning has made apparent-age estimation practical: the DEX model (Rothe et al., ICCV 2015) showed that a convolutional network with an age-group classification head plus a regression head beats either approach alone. UTKFace (20,000+ face images labeled with exact ages, Zhang et al.) provides the training data. This project fine-tunes a ResNet-18 backbone in the DEX style on UTKFace and serves it through a Flask demo application that returns a predicted age with an age-group confidence bar.

03. Problem Statement

There is no accessible, honest tool that estimates age from a face photo and states its own uncertainty — most demos either hide their error or overclaim. The project addresses this by building a two-headed CNN (classification plus regression), evaluating it with mean absolute error on a held-out split, and documenting dataset bias and per-age-band error openly in the report.

04. Objectives

- Fine-tune a ResNet-18 backbone (ImageNet transfer learning) on UTKFace with an age-group classification head plus an age regression head.
- Reach a design target of ~5-6 years MAE on the held-out split, with the training notebook logging MAE and loss curves during the build.
- Compute per-age-band error analysis during the build to document where the model is strongest and weakest.
- Build a Flask demo app that detects and aligns faces and returns the predicted age with an age-group confidence bar.
- Deliver the complete student kit: source code, trained .pth weights, report, PPT and viva Q&A document.

05. Proposed System / Working

The system has three stages. (1) Preprocessing: UTKFace images are split with age-stratified sampling; faces are detected and aligned with a DNN-based detector and cropped to a standard size. (2) Model: a ResNet-18 backbone pre-trained on ImageNet is fine-tuned with two heads — an age-group classifier and an age regressor — optimized with a combined loss; MAE on the held-out split is logged every epoch. (3) Demo application: the Flask app detects and aligns the face in each uploaded photo, runs inference, and displays the predicted age with an age-group confidence bar.

06. Technologies / Components Used

- Python 3.10, PyTorch (training, validation, inference)
- ResNet-18 backbone via torchvision (ImageNet pre-training, fine-tuning)
- OpenCV, DNN face detector (detection, alignment, cropping)
- NumPy, scikit-learn (MAE metrics, per-group analysis)
- Matplotlib, Seaborn (error curves, MAE by age band)
- Flask demo web app with photo-upload UI
- UTKFace dataset (20,000+ face images with age labels)

07. Key Features

- DEX-style two-headed CNN: age-group classification plus regression
- Face detection and alignment preprocessing for consistent inputs
- Flask demo returning predicted age with age-group confidence bar
- Training notebook logging MAE on the held-out split and loss curves
- Per-age-band error analysis computed during the build
- Exportable .pth weights for deployment-style demos

08. Applications / Use Cases

- Age-demographics research demos (prototype scope)
- Teaching material for regression vs classification, MAE, transfer learning and dataset bias
- Reference build for other facial-attribute estimation tasks (emotion, accessories)
- Baseline for comparing newer age-estimation architectures

09. Results & Scope

The deliverable is a design-target-driven build, not a pre-measured product. The training notebook computes mean absolute error on the held-out split every epoch and breaks it down by age band; the report presents these as the build's measured results. The design target is ~5-6 years MAE. Scope is estimation for the most prominent detected face in a photograph — several years of error is normal, accuracy

degrades outside UTKFace's demographic coverage, and this is an educational prototype that must never be used for identity verification, age-gating or legal decisions.

10. Conclusion

Face Age Estimation using CNN delivers a complete applied deep-learning project: a real perception problem, the public UTKFace dataset, a DEX-style two-headed ResNet-18 trained through a fully logged experiment, and a demo app that makes the result visible in seconds. Its honest treatment of error and bias — MAE by age band, documented skew — gives the student defensible, viva-ready material.

11. References

- Rothe, R. et al. — DEX: Deep EXpectation of Apparent Age from a Single Image (arXiv:1505.01814, ICCV 2015), the two-headed approach this build follows.
- Zhang, Z. et al. — UTKFace dataset (introduced in Age Progression/Regression by Conditional Adversarial Autoencoder, CVPR 2017), the 20,000+ age-labeled face images used for training.
- He, K. et al. — Deep Residual Learning for Image Recognition (arXiv:1512.03385, CVPR 2016), the ResNet backbone used in the build.
- PyTorch documentation (pytorch.org) — training and transfer-learning tooling used in the build.