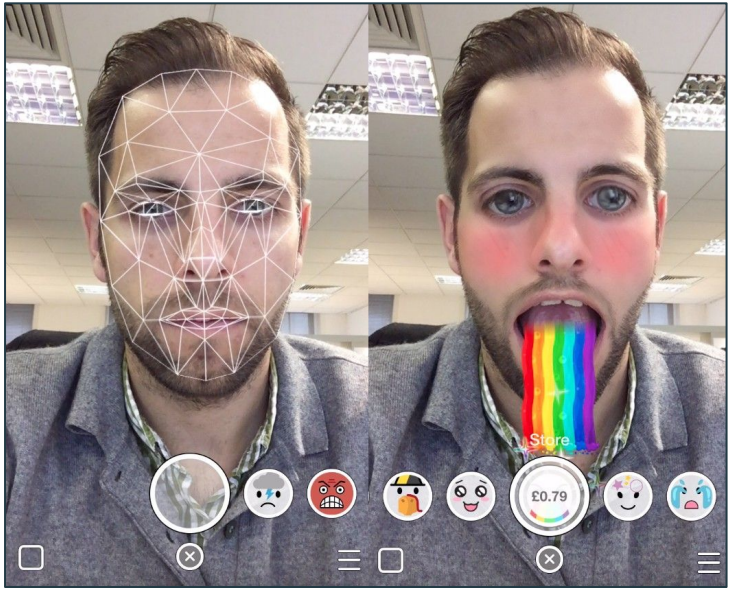
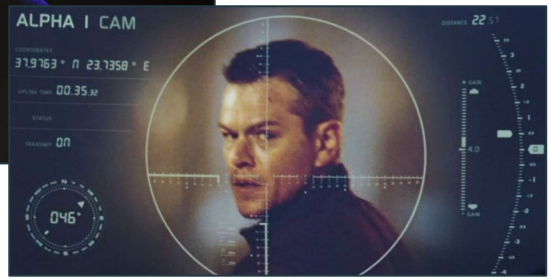
A Simple Facial Recognition System

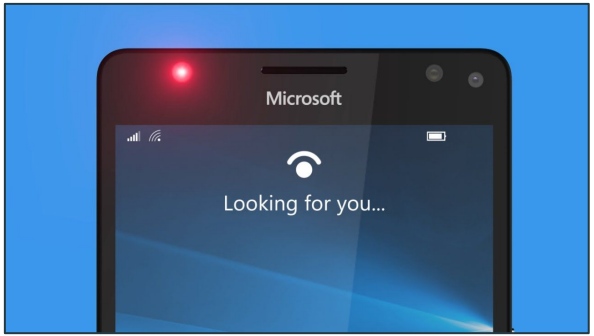
Jeremy Midvidy, jam658
Final Project Presentation

EECS 332: Computer Vision
Fall 2018, Northwestern University

Facial Recognition





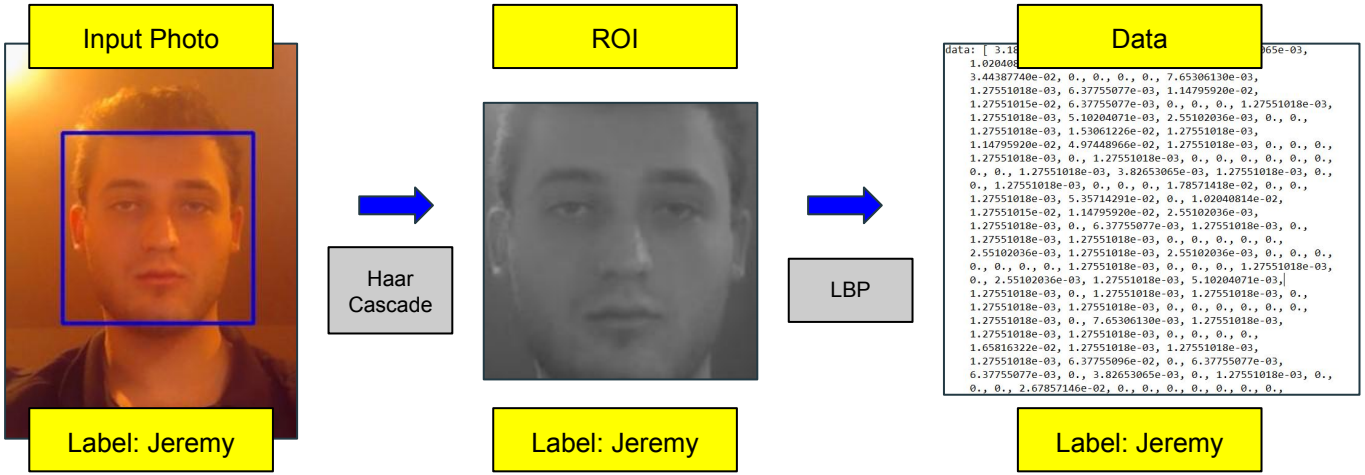


A Simple Facial Recognition System

- Part 1: Build database of “verified” Faces
  - Like setting up your FaceID on your iPhone
  - Like setting up access to a secure facility
- Part 2: Train the Facial Recognition System to be able to classify inputs
  - A person claims to be someone in the database
  - FR System → take pictures of that person’s face and say:
    - YES → you ARE that person → allow access
    - NO → you are NOT that person → deny access

Part 1: Build Database

- Use OpenCV's *haar cascades* to extract a *region of interest* from each input photo
- Use *regions of interest* and known *labels* to train a recognizer
  - Recognition uses *LBP* to build feature model of a given label



Part 2: Train Classifier

- Use input images (150) per person to aggregate data for each label
- Given an input image, use *LBPClassification* to determine classification label and *confidence*
  - If confidence is above learned threshold → ACCEPT
  - Below learned threshold → REJECT
- Learn threshold:
  - Test known images (e.g. Jeremy) against itself → get confidence
  - Threshold = Average of Known Labels + 20 (wiggle room)
    - Typically, threshold is ~50
- Process inputs same as building database:
  - Picture, ROI, Data → Classifier → Accept/Reject

DEMO

Future Improvements

- More robust input images (only front-profile with no fashion accessories on head)
- Use more robust classification system with more features (eyes, cheeks etc…)
  - Deep Learning with TensorFlow
  - 3D Facial Image reconstruction
  - LIDAR detection
- This implementation is far from the one found on an Iphone X!