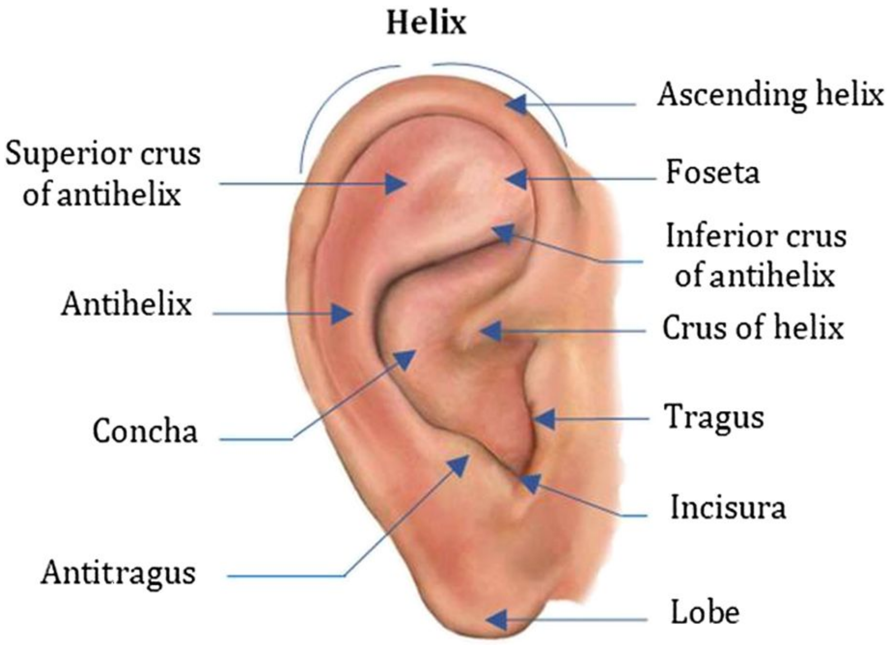
# Eardentity: An Android App for Ear-based Identification

Giorgos Karatziolas, Tu Lan, Max Stenham, Adam Piekarski, Ethan Harris

# Contents

# Introduction: Ear as an Identifier

- The unique curves of the ear make it suitable for a biometric identifier.
- Ears only scale as a person ages.
- It's often all that can be found after an accident!
- Integration into a app allows ears to be used for personal security.



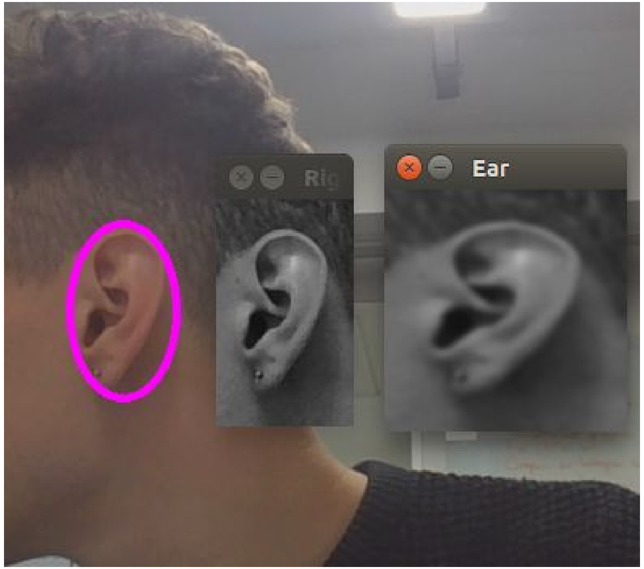
# Algorithm Overview

3 stages:

1. Ear detection:  Haar cascade [1]
2. Feature extraction & description: LBP histograms [2][3]
3. Recognition: chi-square test [3]

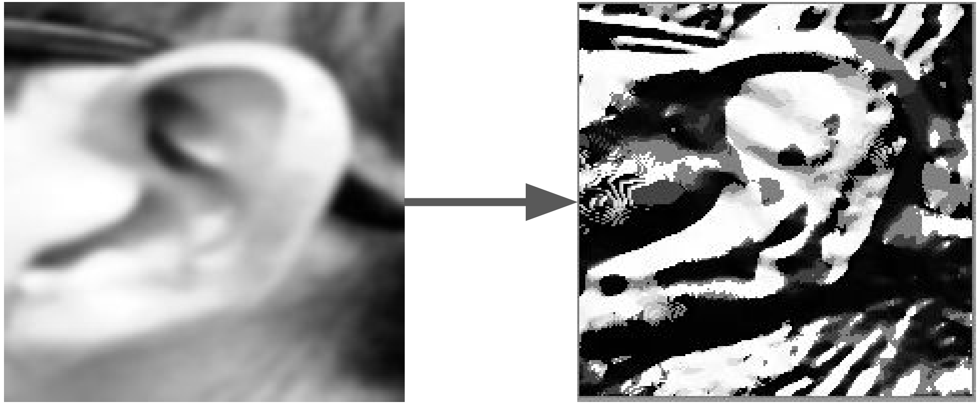
# Finding an Ear in a Scene

- Separate Haar cascades for left and right ears are included in OpenCV.
- OpenCV provides a native method for running Haar cascades over images.
- Isolated ear is then set to a standard size and blurred.



# LBP for Ear Description

- Extended LBP with radius = 2 and neighbours = 16
- Tuned parameters include:
  - Image size
  - Image blur
  - LBP radius
  - LBP neighbours
  - LBP image segmentation size

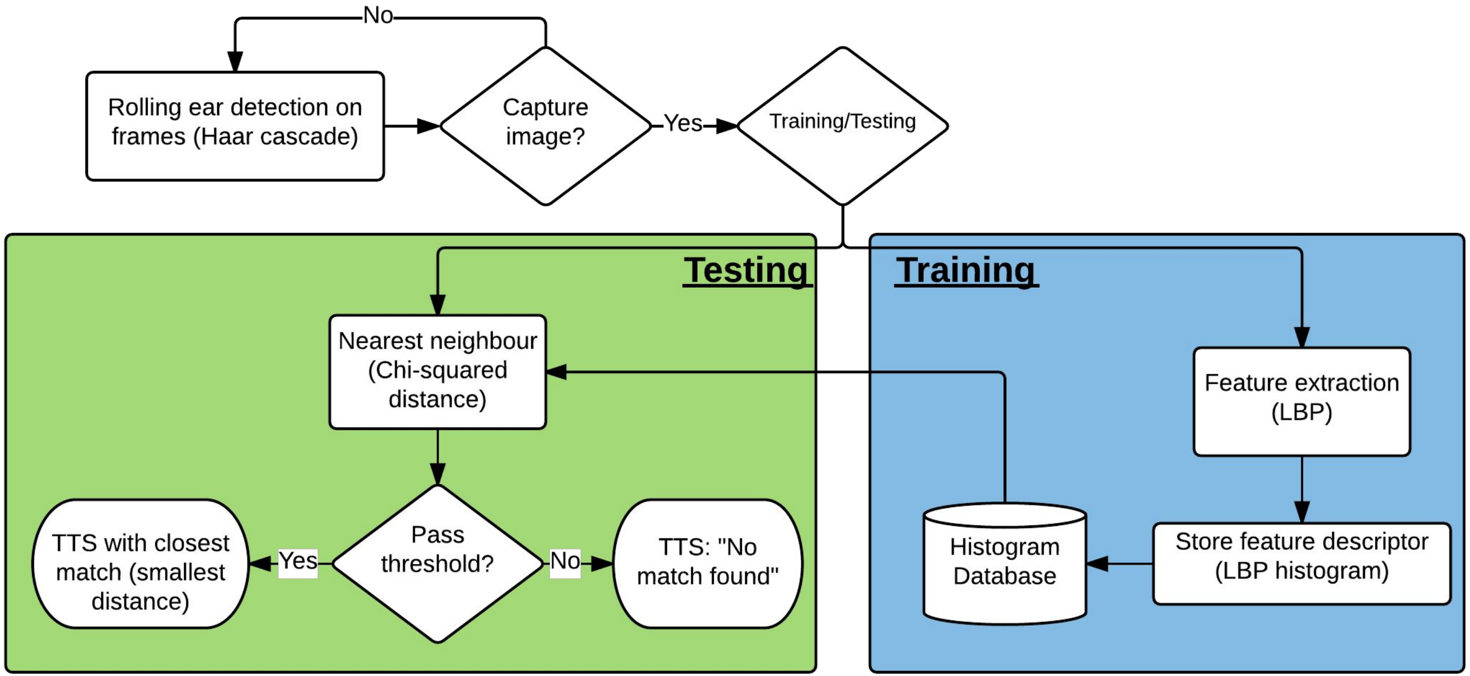


# Comparing LBP Histograms

- Compute the histograms with window size 20x20 using non-uniform bins.
- Concatenate the histograms from each window.
- Calculate chi-square distance between detected image and those in the database.
- The smallest distance is the best-guess for a match

$$\chi^2(X,Y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{(x_i + y_i)}$$

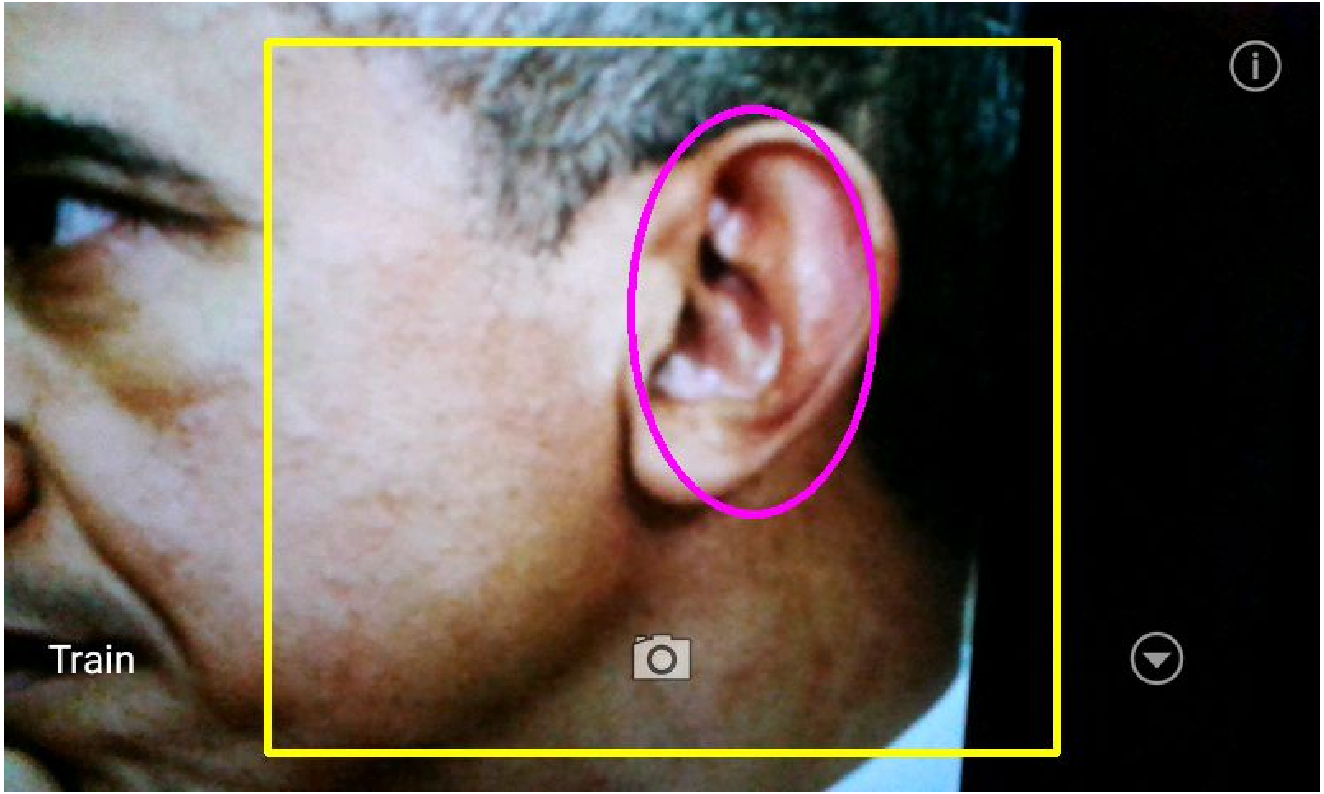
# Application Operation



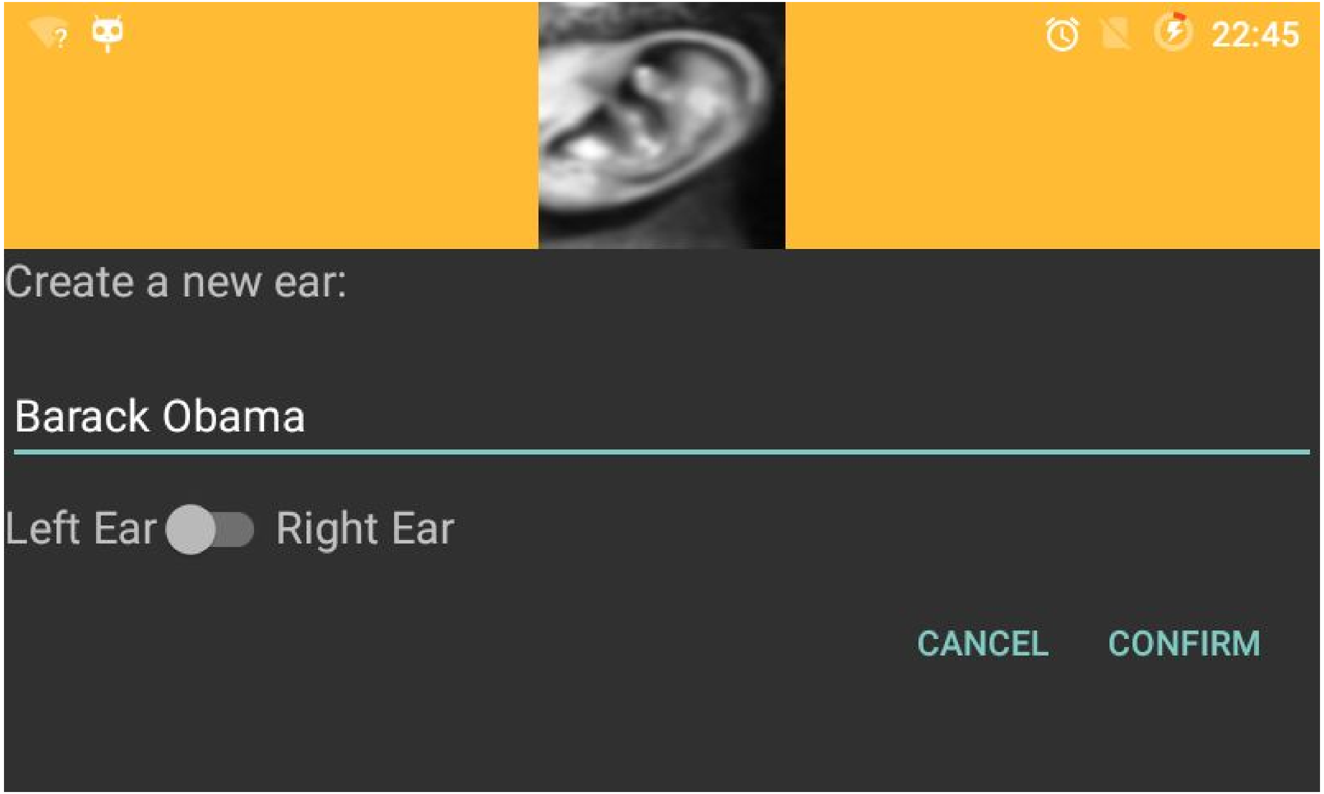
# Implementation

- Android
  - User interface
  - Camera access
  - Data persistence (sqlite and saving descriptors)
- OpenCV
  - Image processing
- Java Native Interface
  - Perform resource-intensive image processing

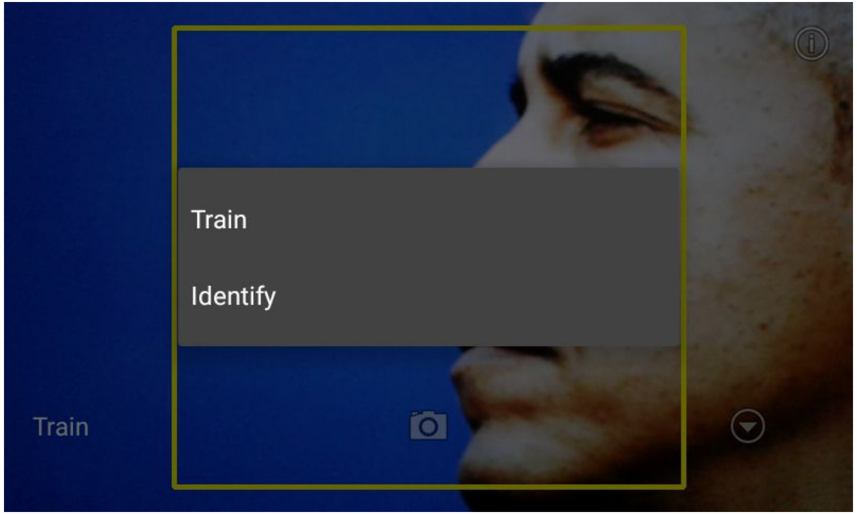
# Training and Tracking

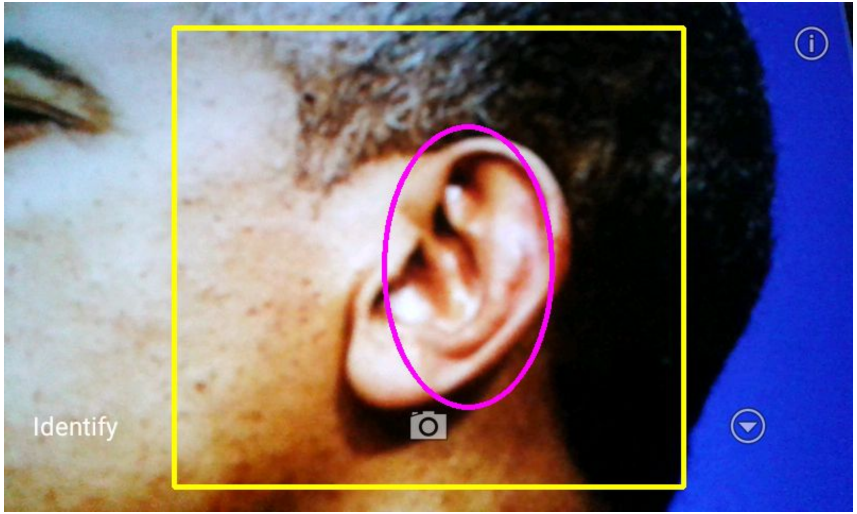


# Adding Ears to the App

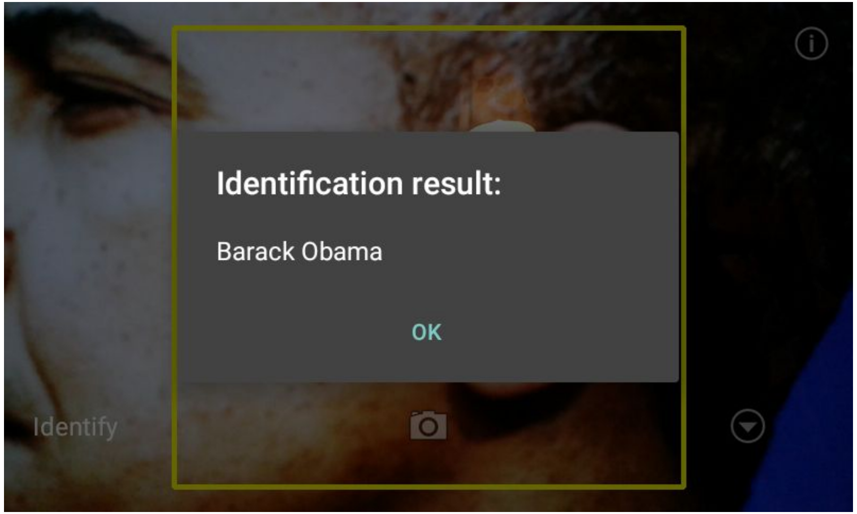


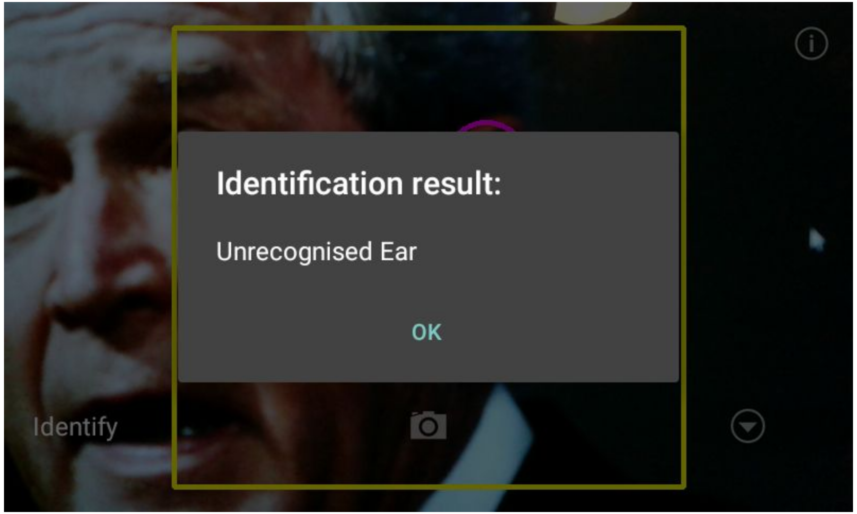
# Identifying ears



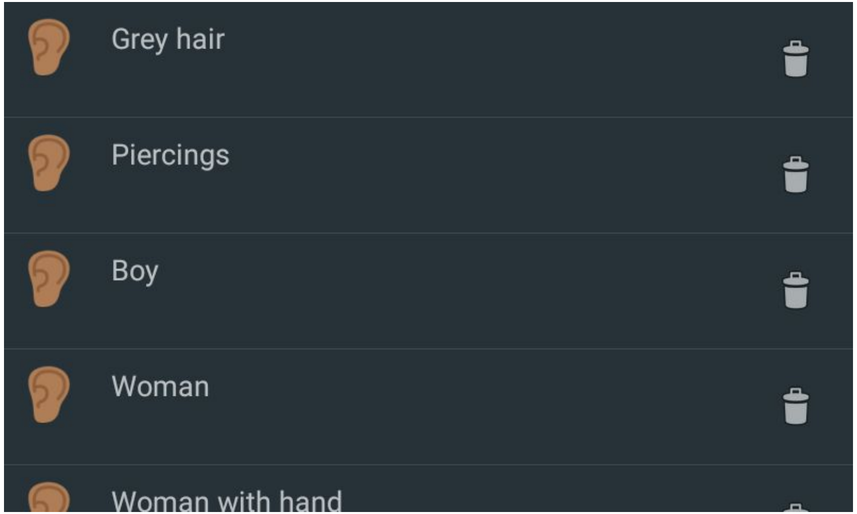



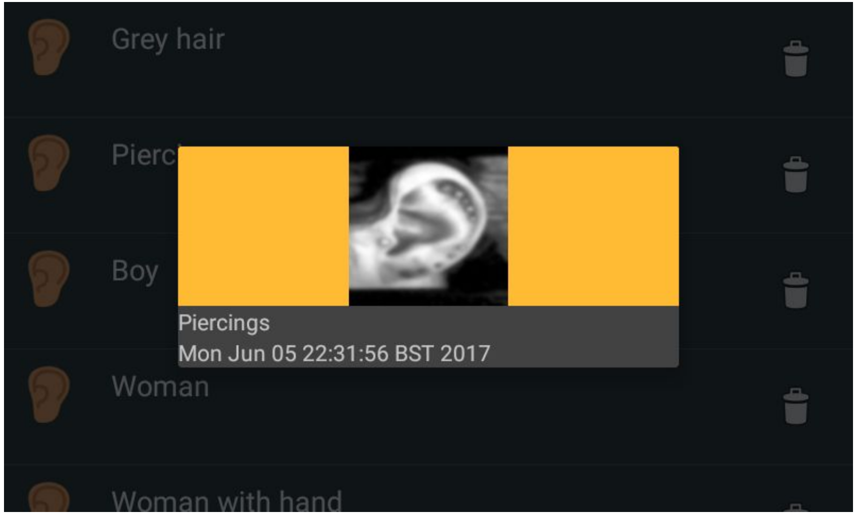
# Identifying ears



# Managing ears



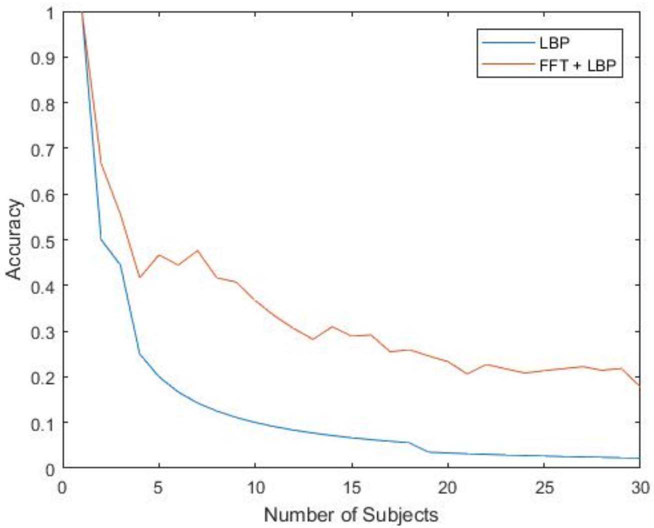



# Demonstration

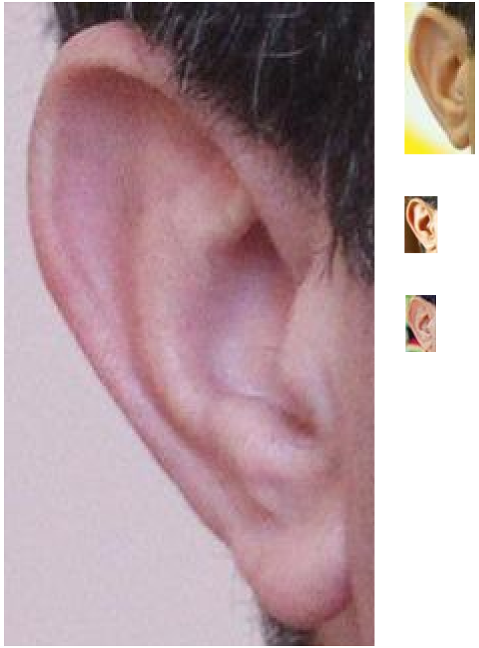
http://fatcookies.github.io/misc/openCVSamplefacedetection-debug.apk

# Testing & Results

- Haar cascades give effective ear detection under highly variant lighting
- Identification performance on Annotated Web Ears (AWE) Dataset [5]



# AWE Dataset Issues



Full resolution, right ears for subject one

- Dataset images have extremely high variance
- Images in app much more consistent
  - Same resolution
  - Same ear angle (due to Haar Cascade)
  - Same exposure

# Evaluation

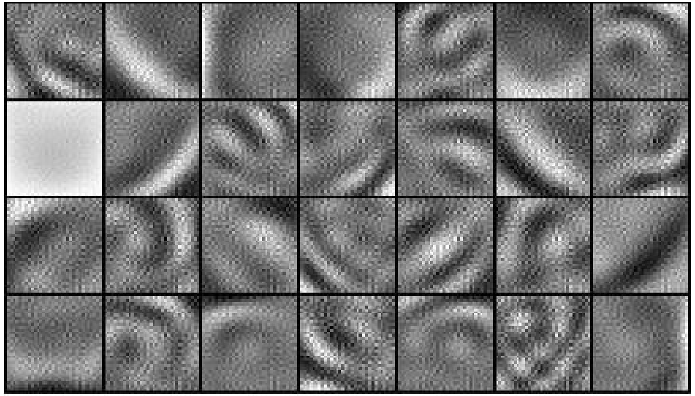
- Real-time ear detection
- High accuracy on small datasets
- High light invariance

# Reconstruction Independent Component Analysis (RICA)

- Learn overcomplete, sparse basis, W, for the data:

$$\min_{W} \quad \lambda \left\| Wx \right\|_1 + \frac{1}{2} \left\| W^T W x - x \right\|_2^2$$

- Obtain filters from weights:
  - Force Field Transform on images in [5]
  - Filters correspond to shape / edge descriptors
  - Noisy on un-whitened data
- Use as filters for Conv. layer of CNN
  - Fast - only have to train SoftMax on new images
  - Low accuracy with one ear image per subject
  - Faster than traditional CNN but too slow for mobile



# Other Methods: Force Field Transform [4]

Objective: smooth image while highlighting features

Method: apply theory of gravitational attraction to image

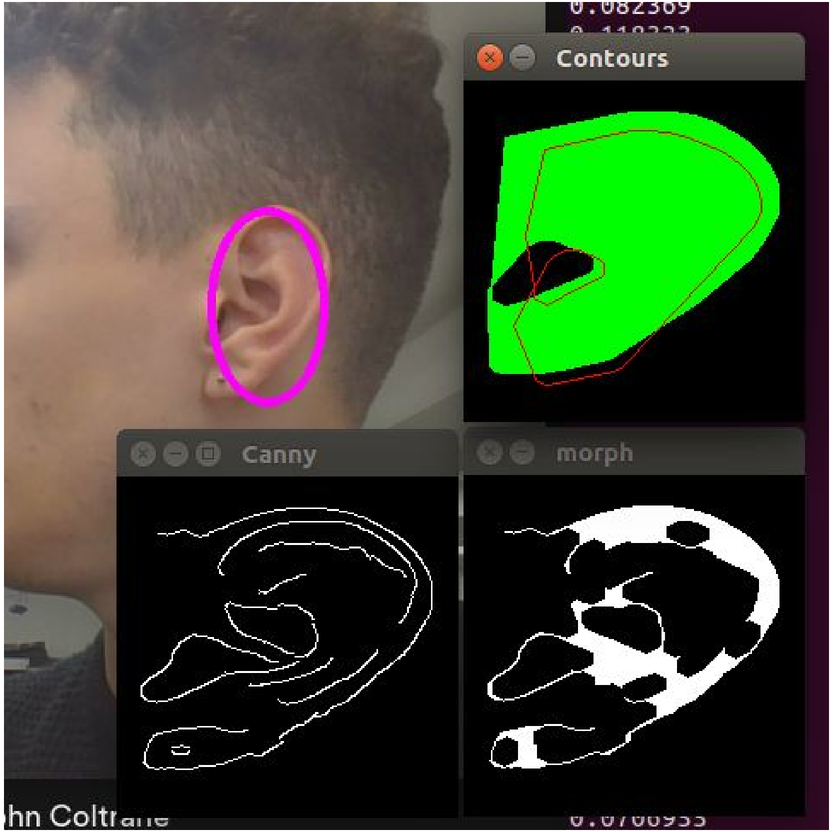
- Transform applied through convolving image with centred unit force field
- For speed: (crucial requirement)
  - Convolution in spatial domain ≡ multiplication in frequency domain

Issues:

- Edge effects + LBP
- Noise resistance unnecessary
- Slow (even in Fourier domain)

# Other Methods: Contour Matching

- Contour based matching was attempted.
- Canny contours could be collected quickly.
- But they were highly variant with light.
- And gave a low matching accuracy.



# Conclusion

- Produced ear biometric application
- Explored multiple methods
  - Chose best performing

Further work:

- Improve feature extraction (Uniform LBP, force field transform)
- Explore other matching algorithms and distance metrics
- Application improvements (both ears)

# References

[1] Feature detection method:

Modesto Castrillón-Santana, Javier Lorenzo-Navarro, and Daniel Hernández-Sosa. 2011. An study on ear detection and its applications to face detection. In *Proceedings of the 14th international conference on Advances in artificial intelligence: spanish association for artificial intelligence* (CAEPIA'11), Jose A. Lozano, José A. Gámez, and José A. Moreno (Eds.). Springer-Verlag, Berlin, Heidelberg, 313-322.

[2] Feature description method:

A.Abaza and M. Harrison, “Ear recognition: a complete system”, Cairo University, Cairo, Egypt, 2015

[3] LBP and chi-square test implementation from Bytefish:

https://github.com/bytefish/opencv/tree/master/lbp

[4] Force Field Transform:

J. Hurley, M. Nixon, J. Carter, “Force field feature extraction for ear biometrics” in *Computer Vision and Image Understanding 98* (2005), pp 491–512

[5] Dataset for testing:

Emeršič, Žiga, Vitomir Štruc, and Peter Peer. "Ear Recognition: More Than a Survey." arXiv preprint arXiv:1611.06203 (2016).

# Credits

- Giorgos - Feature extraction
- Lan - Matching
- Max - Force Field Transform
- Adam - Android development
- Ethan - Matching

# Thank you