

Through the eyes of RoboFace

Robotics Practical – Advanced Lab

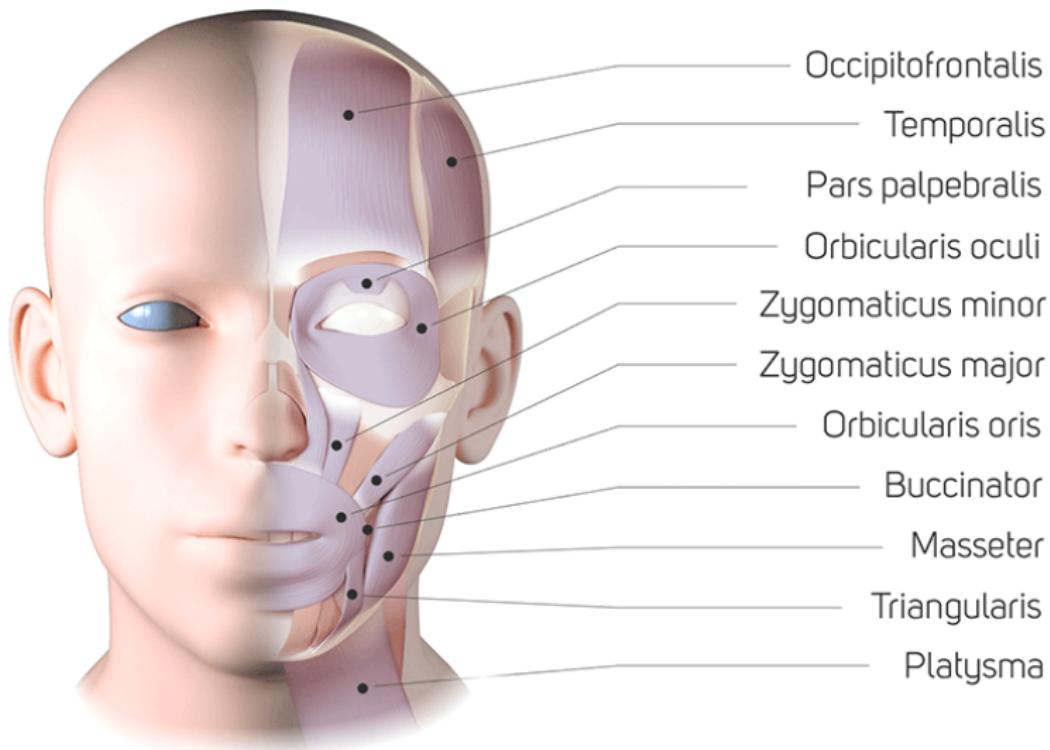
Kevin Kiefer, Letiția Elena Pârcălăbescu

the 28th of March 2017

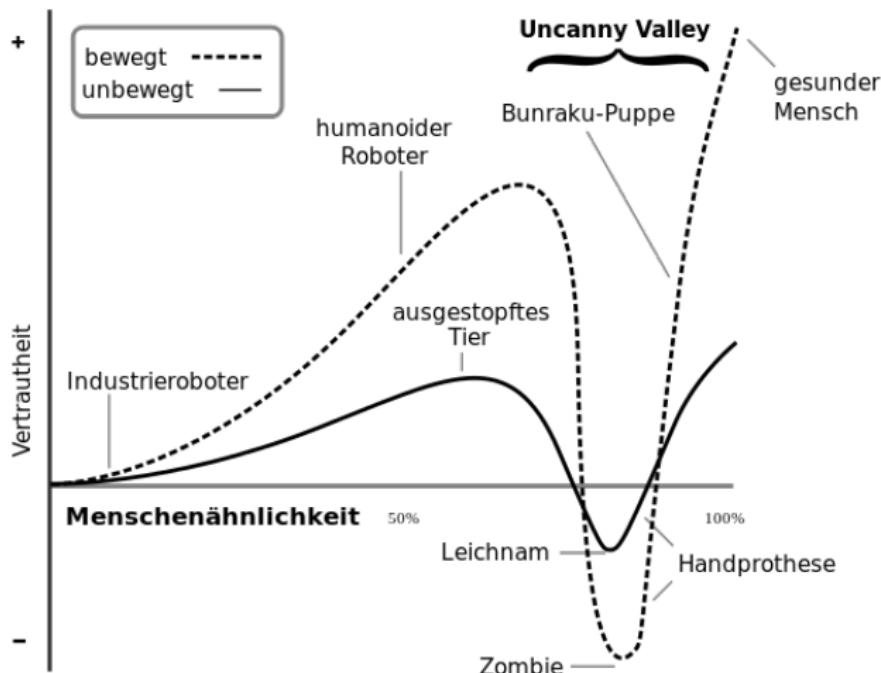
Structure

- 1 Introduction
- 2 Hardware
- 3 Software
- 4 Problem
- 5 Dataset
- 6 Own Approach
- 7 Outlook and Refs.

HumanFace



Uncanny Valley

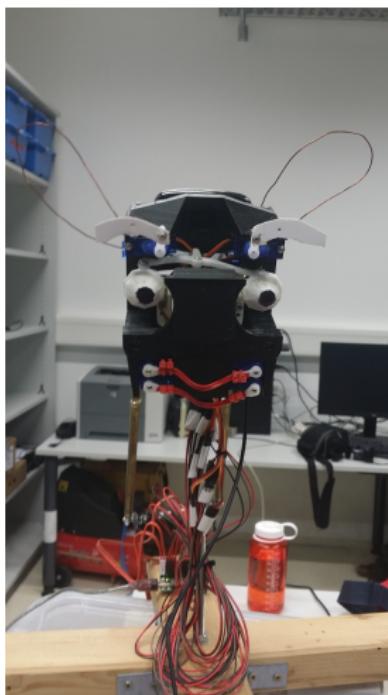


Uncanny Valley



RoboFace

Hardware



RoboFace

Hardware



RoboFace

Software

- ① dependecies: pololu-usb-sdk, RapaPololuMaestro, boost(python), python(2 or 3)
- ② low-lvl code implemented in c++
- ③ python api build on top
- ④ main workflow implemented in python

RoboFace

Software

```
template<size_t N>
class ServoConfig
{
public:
    ServoConfig(
        const std::array<int,N> & servoChannel ,
        const std::array<int,N> & servoPos
    );
    // getter & setter
private:
    std::array<int, N> servoChannel_;
    std::array<int, N> servoPos_;
};
```

RoboFace

Software

```
template<size_t NUM_SERVOS>
struct ServoConstraints
{
    public:
        ServoConstraints(
            int min, int max,
            const std::array<int, NUM_SERVOS> & channels
        );

        bool isValidChannel(int channel) const;
        bool isValidPosition(int pos) const;

        // isValidChannelArray & isValidPositionArray

        template<size_t N>
        typename std::enable_if<(N <= NUM_SERVOS), bool>::type
        isValidConfig(const ServoConfig<N> & config) const;

    private:
        std::array<int, NUM_SERVOS> channels_;
        int minPos_;
        int maxPos_;
};
```

RoboFace

Software

```
class Face
{
public:
    Face(
        int x_len = 640 , int y_len = 480,
        float x_weight = 0.5f, float y_weight = 0.5f,
        const std::string & dev = "/dev/ttyACM0"
    );

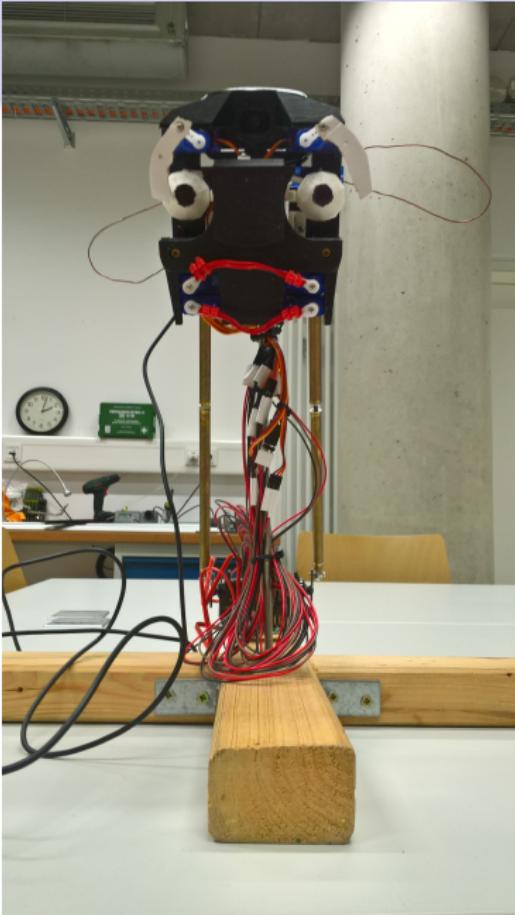
    template<size_t N>
    void applyConfig(const ServoConfig<N> & config)

    void happy(bool moveHead = true)
        // more emotions

    void moveHeadX(int x)
    void moveHeadY(int y)
    void moveHead(int x, int y)
private:
    // ...
};
```

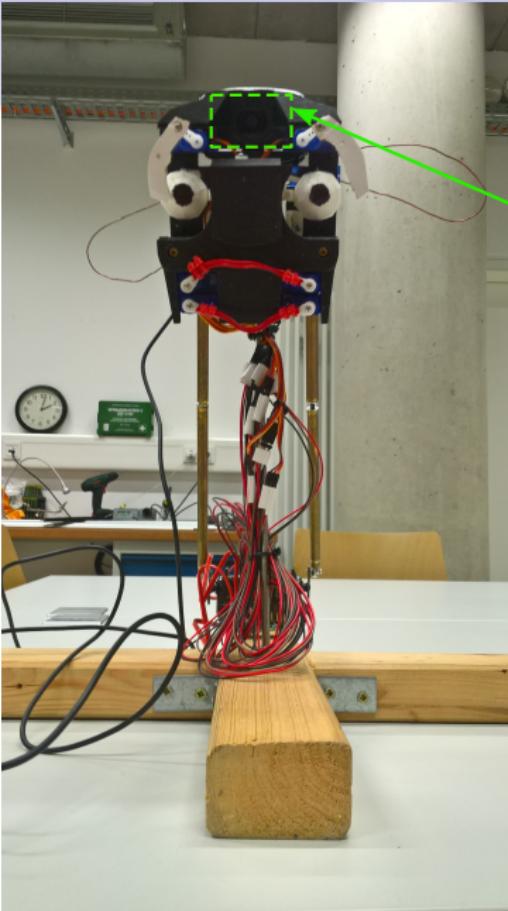
RoboFace

Deep Vision Problem



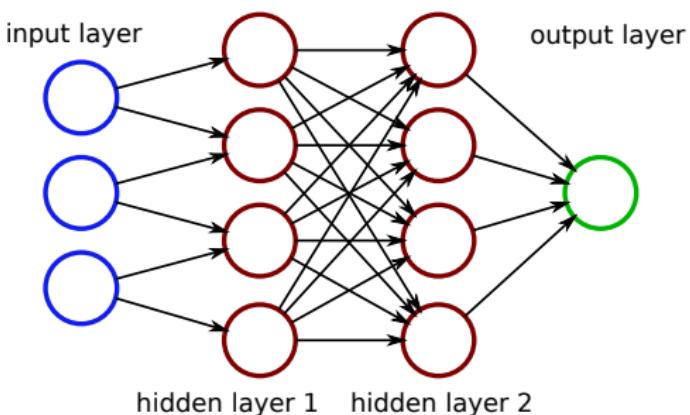
RoboFace

Deep Vision Problem



Deep Vision

Short Intro into Neural Networks



Deep Vision

Short Intro into Neural Networks

A neuron is a linear function, followed by nonlinearity.

$$f\left(\sum_i w_i x_i + b\right)$$

Want to find the weights w_i .

Universal Approximation Theorem:

- ① 1 hidden layer: continuous functions
- ② 2 hidden layers: ANY function!

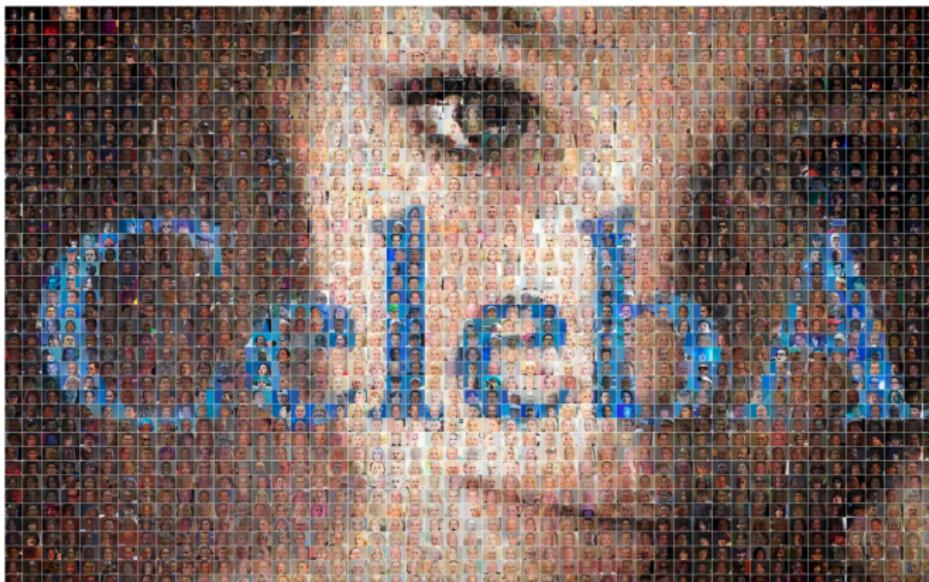
CelebA Dataset web (2016)

202.599 images, 10.177 identities, 40 attributes per image

Large-scale CelebFaces Attributes (CelebA) Dataset

Ziwei Liu Ping Luo Xiaogang Wang Xiaoou Tang

Multimedia Laboratory, The Chinese University of Hong Kong



CelebA Dataset

Sample Images

Eyeglasses



Wearing Hat



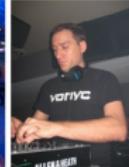
Bangs



Wavy Hair



Pointy Nose



Mustache



Oval Face



Smiling



CelebA Dataset

Attribute Selection

chose 13 labels out of 40 available

- | | |
|-----------------------|--------------------|
| ① Black Hair | ⑧ No Beard |
| ② Blond Hair | ⑨ Smiling |
| ③ Brown Hair | ⑩ Straight Hair |
| ④ Eyeglasses | ⑪ Wavy Hair |
| ⑤ Gray Hair | ⑫ Wearing Earrings |
| ⑥ Male | ⑬ Wearing Lipstick |
| ⑦ Mouth Slightly Open | |

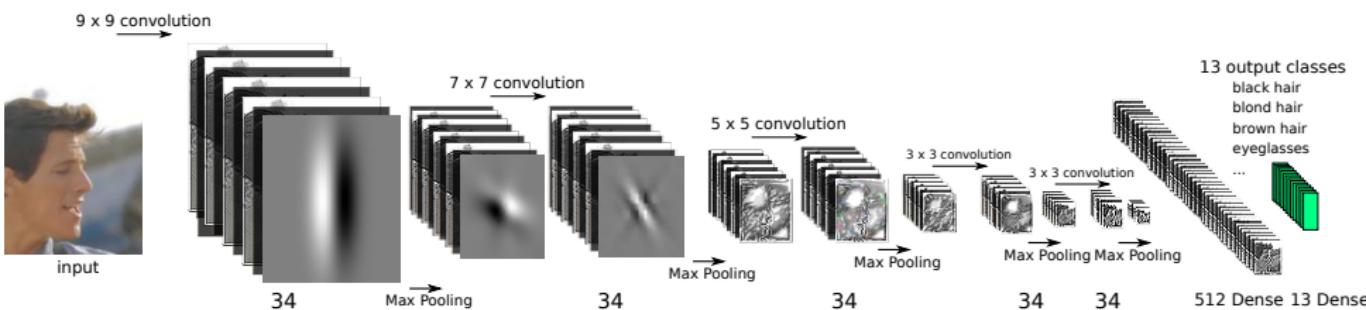
Own Approach

Architecture

- ① 32 of neurons in each convolutional layer
- ② Activations: Relu
- ③ 9 x 9 convolution → Max Pooling
- ④ 7 x 7 convolution → Max Pooling
- ⑤ 5 x 5 convolution → Max Pooling
- ⑥ 3 x 3 convolution → Max Pooling
- ⑦ 3 x 3 convolution → Max Pooling
- ⑧ Dropout(0.25)
- ⑨ 512 Dense
- ⑩ Dropout(0.5)
- ⑪ 13 Dense
- ⑫ Sigmoid
- ⑬ Binary Crossentropy
- ⑭ Adadelta optimiser
- ⑮ Overall: 125,709 parameters

Own Approach

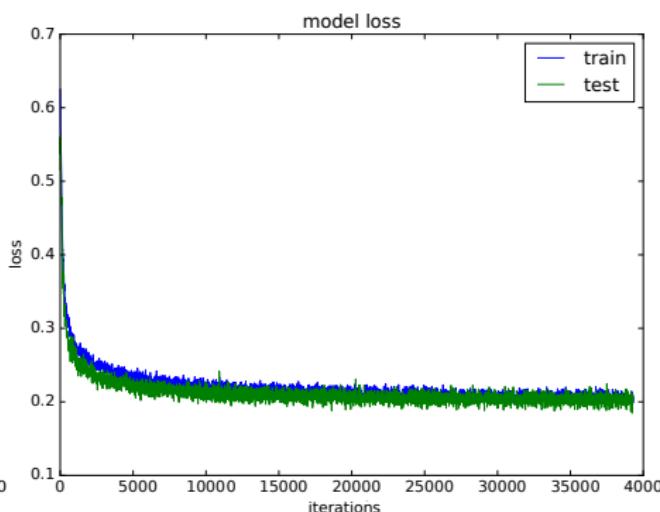
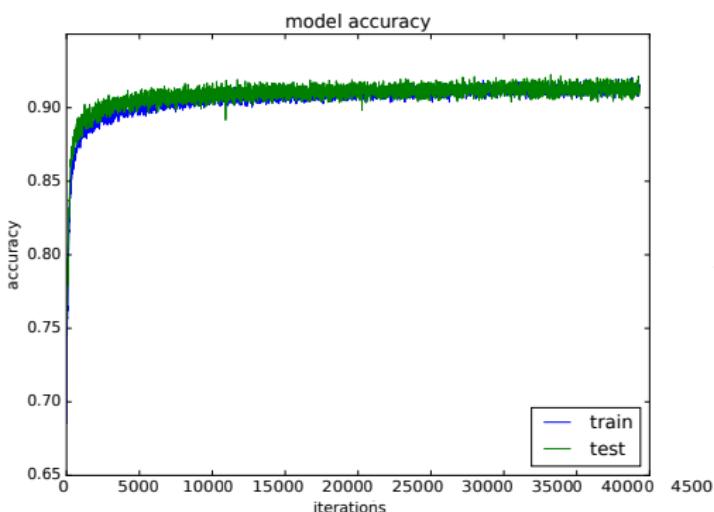
Architecture



Own Approach

How to not do it!

Without data normalisation: beautiful training curve, BUT...



Own Approach

How to not do it!

... BUT no generalisation capability at all!

The predicted classes on the real world images from the robot are:

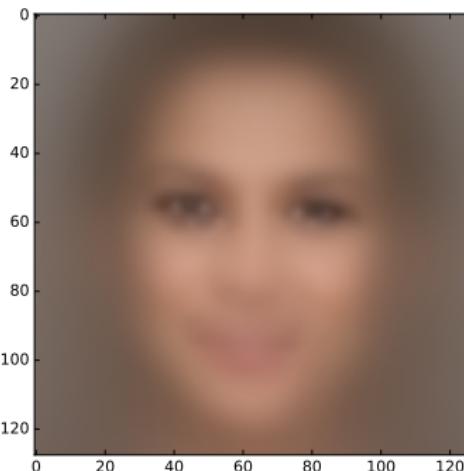
Male (0.89), No beard (0.76)



Data normalisation

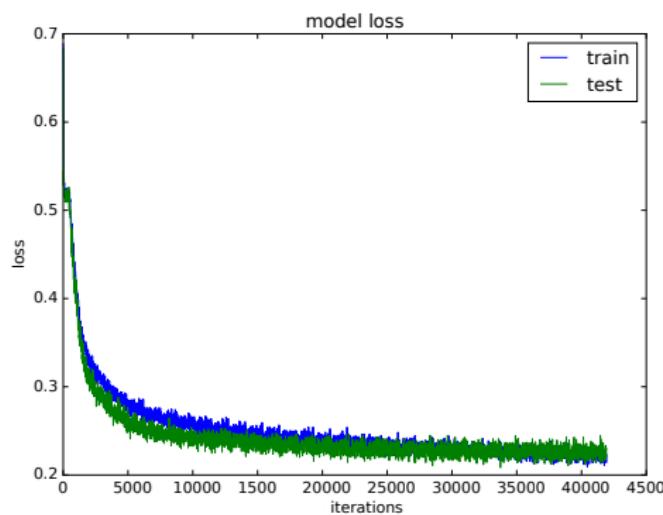
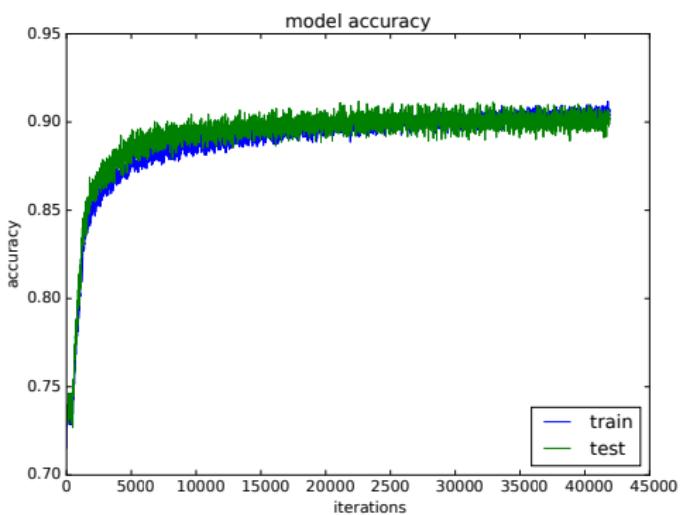
How to really do it!

- ① center the eyes
- ② resize images to have the same inter ocular distance
- ③ rotate image to make the inter ocular line look horizontal
- ④ resize image to 128 x 128 pixels
- ⑤ subtract the mean face



Own Approach

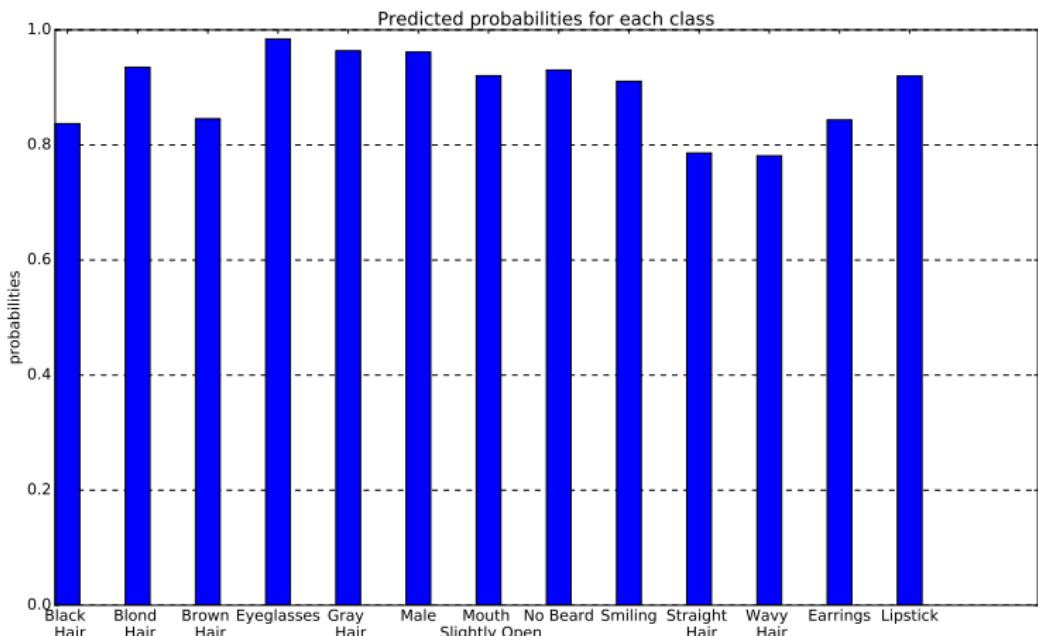
Training Curves



Own Approach

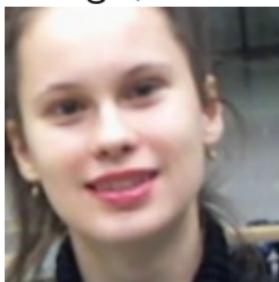
Final Accuracy per Class

overall accuracy of 90%

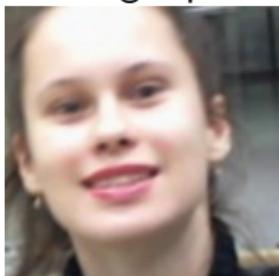


Exemplary Results

'Black Hair', 'Mouth Slightly Open', 'No Beard', 'Smiling',
'Wearing Earrings', 'Wearing Lipstick'



'Black Hair', 'Mouth Slightly Open', 'No Beard', 'Straight Hair',
'Wearing Lipstick'

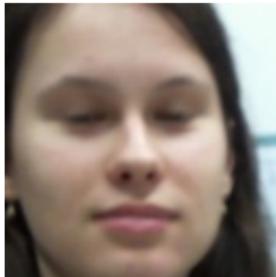


Exemplary Results

'Black Hair', 'Mouth Slightly Open', 'No Beard', 'Smiling',
'Straight Hair', 'Wearing Earrings', '**Wearing Lipstick**



'Black Hair', '**Male**', 'Straight Hair'



Demo time!



Outlook

- hardware improvements
- speech recognition
- data acquisition

References

Celeba website, 2016. URL

<http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html>.

GitHub: <https://github.com/LetiP/RoboFace>