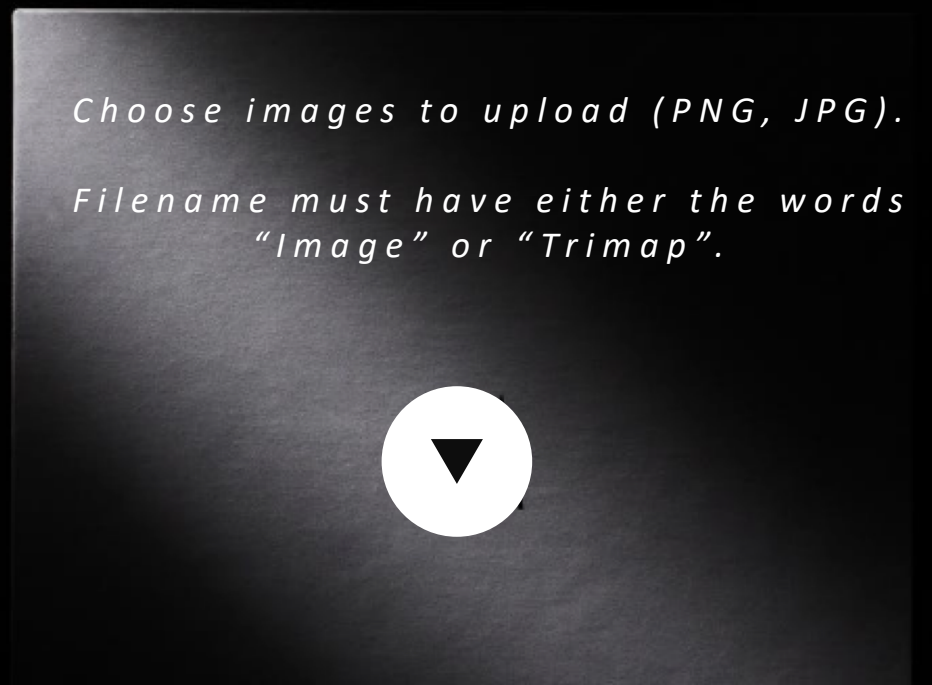
WE ARE DEEPMASQUE

We Create Real Masks

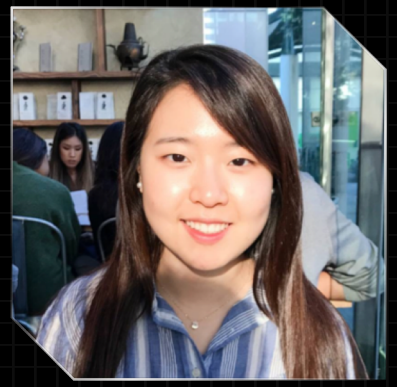


TEAM DEEPMASQUE

Interdisciplinary team of data scientists

Data Scientist
Yu-Chieh Lee

- Bachelor of Sci. in EECS
- Incoming Software Engineer at LinkedIn

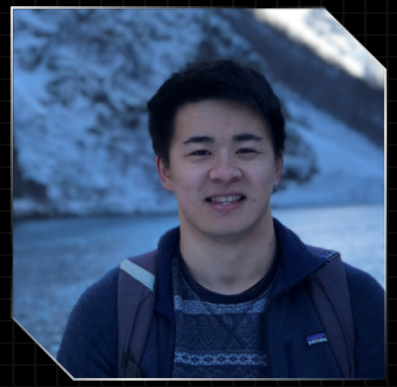


Data Scientist
Sierra Park

- Bachelor of Art in Applied Math
- Incoming Data Analyst at Wayfair

Data Scientist
Leo Li

- Bachelor of Art in Cognitive Science
- Bachelor of Art in Statistics



Data Scientist
Chris Chin

- Bachelor of Sci. in Civil Engineering
- Incoming Researcher at MIT

THE GREEN SCREEN TECHNIQUE

A ubiquitous post-production technique

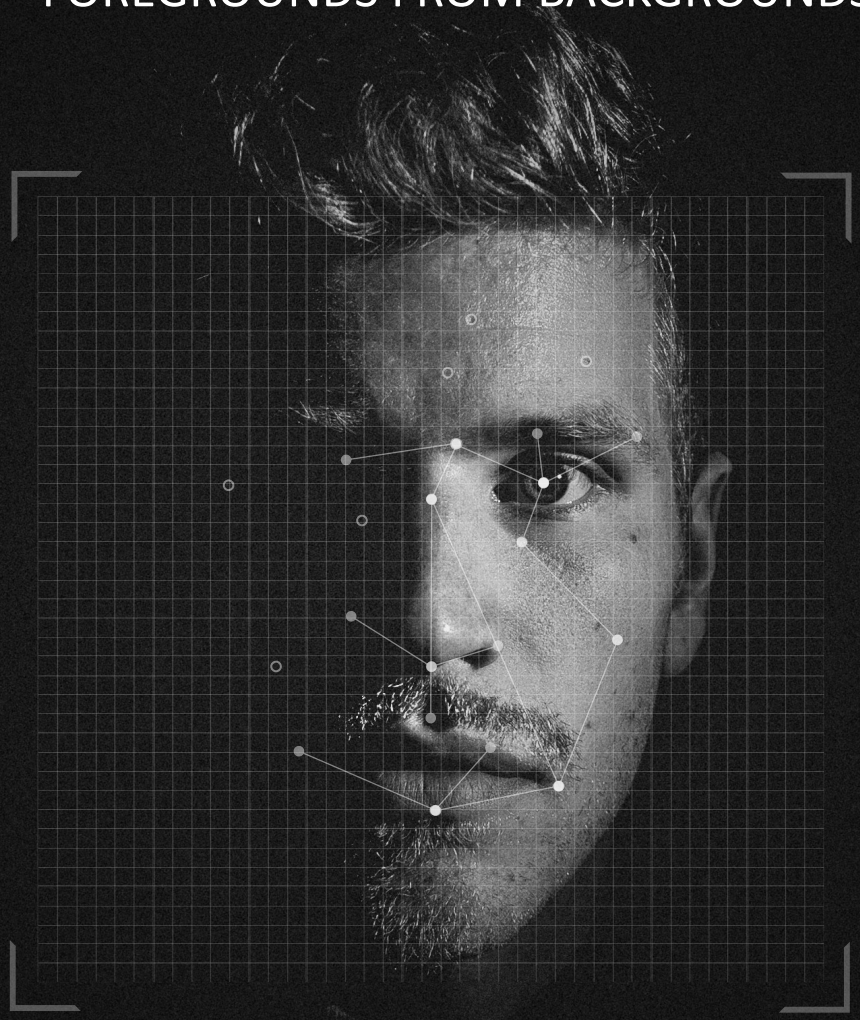
- A green screen is a uniform backdrop used in filming, 3-D printing, and VR
- Allows editors to easily separate the subject from the background to change the background
- Used by everyone from amateur film-ers to Hollywood producers

PROBLEMS

EXPENSIVE,
TIME-CONSUMING,
HIGH BARRIER FOR ENTRY

- Green screen film has to be processed by an editor
- Editing film is expensive, time-consuming, and difficult for beginners
- Is there a better way?

WE CREATE ACCURATE MASKS TO DIFFERENTIATE FOREGROUNDS FROM BACKGROUNDS



++++++

+

FACE

Our mission is to automate the editing process of photos in VR, AR, and entertainment, etc., through precise segmentation via deep learning.

Advisor

Richard Berwick

Richard@twindom.com

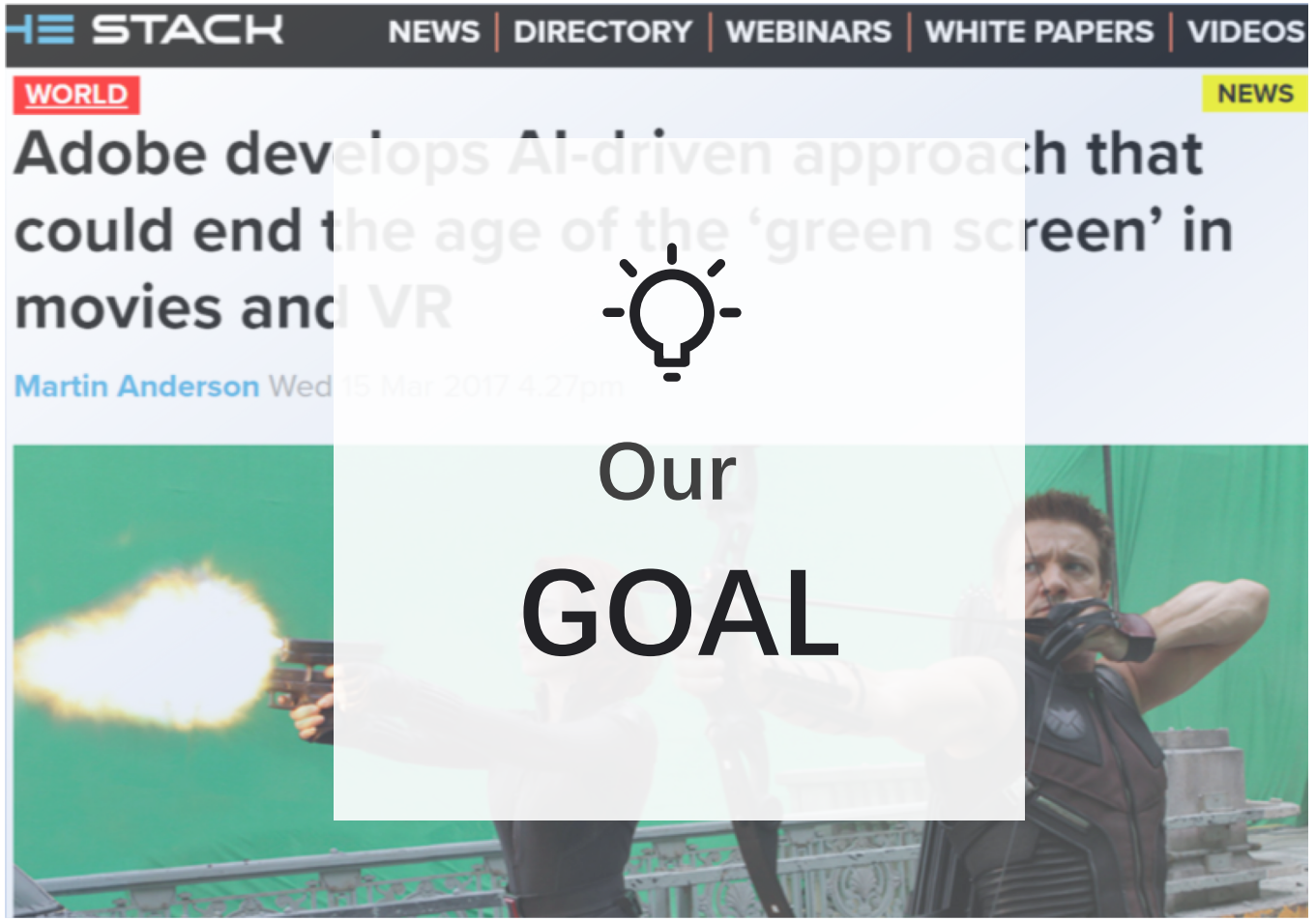
Image, Trimap, Alpha Matte
Portrait FCN+, KNN-Matting
Twindom
>>>>>>>>>>>>>>>>>>>>>>>

2018/5/2
Product Demo

www.deepmasque.com

ELIMINATE the need for a middle-man

- Make website where users can upload a portrait and instantly separate subject and background
- Enable users to change the background
- Film-makers of any calibre can circumvent the green screen process



OUR ALGORITHM vs MARKET COMPETITORS

Images from one-click background removal websites

01 INPUT IMAGE

02 OUR SEGMENTED IMAGE

03 BURNER.BON-ANZA.COM

04 MALABI.CO

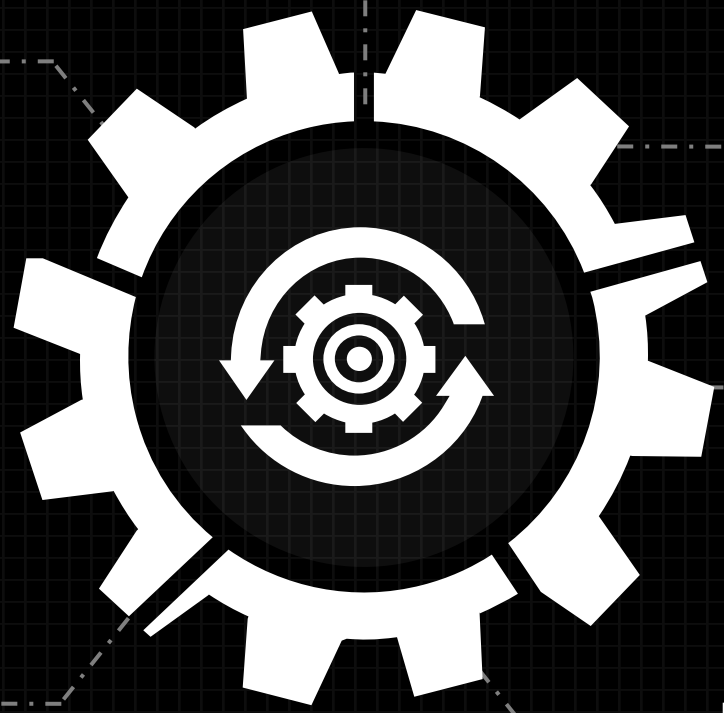
SOLUTION ARCHITECTURE

Pre-processing

- Take in RGB and find facial points by finding offset of x and y components to transform the picture to canonical pose.
- Apply mean mask transform to the image

Portrait FCN+

- Apply VGG on images using multiple smaller convolutional layers/filters for faster computation to capture nonlinearity
- Deconvolution to enlarge the small image back to its original size



Alpha Matte

ResNet

Training deep networks by finding the true y-values by predicting the error from the predicted y-values

Trimap

KNN Matting

User uses K nearest neighbors to get closed-form solution

SOLUTION ARCHITECTURE

Input

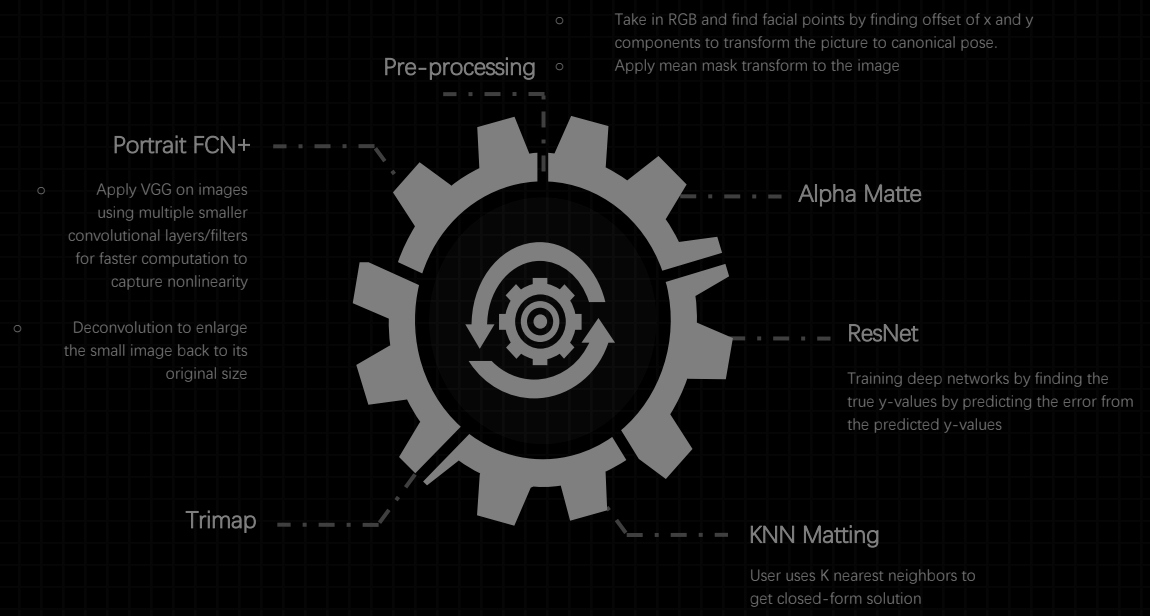
Input image of a person

SERVER 00

Amazon EC2 Server

Output

Foreground & Background Segmented



OVERVIEW OF SOLUTION

Making the complex seamless

01

INPUT IMAGE

02

TRIMAP

03

ALPHA MATTE

04

COMPOSITE IMAGE

Portrait FCN+

(in TensorFlow)

KNN Matting

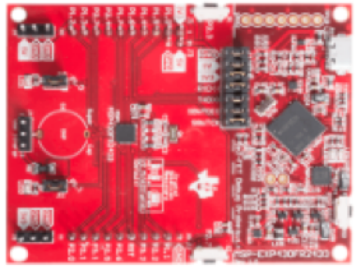
& Residual Network

Applying foreground on

another image

A tremendous memory demand reduction compared to state of the art

HARDWARE RESULTS & DEMAND



MSP-EXP430FR2433

Low cost MSP430 Value Line FRAM MCU LaunchPad.

Featuring the MSP430FR2433: 16-bit MCU with 16KB FRAM, 4KB SRAM and 10-bit 200KSPS ADC.

20-pin LaunchPad kit standard leveraging the BoosterPack ecosystem.

2 buttons and 2 LEDs for user interaction.

Capacity: 4 KB

Space Gray

Capacity: 64-256GB

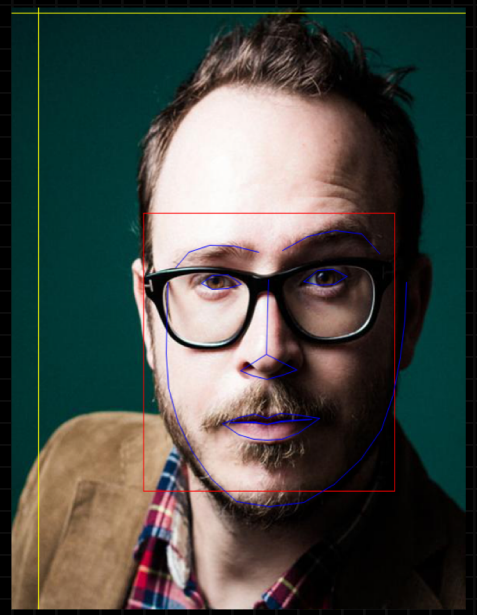
Reduced memory demand by over 2 GB (>> 2 times) with the elimination of DIM acting as middleman between trimap and alpha matte.

FYI: KB = 10^3, GB = 10^9

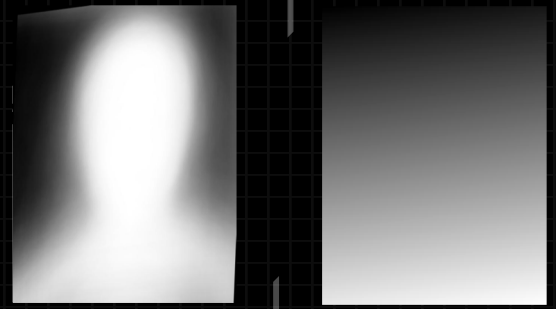
In fact, our trimap to alpha matte algorithm works particularly well due to PortraitFCN+ extreme accuracy in trimap generation and can be implemented on a LaunchPad.

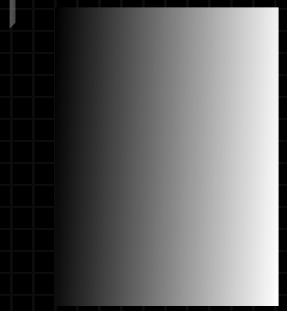
PORTRAIT FCN+: PREPROCESSING

01 INPUT IMAGE



02 DEEP METRIC LEARNING: FACIAL FEATURIZATION

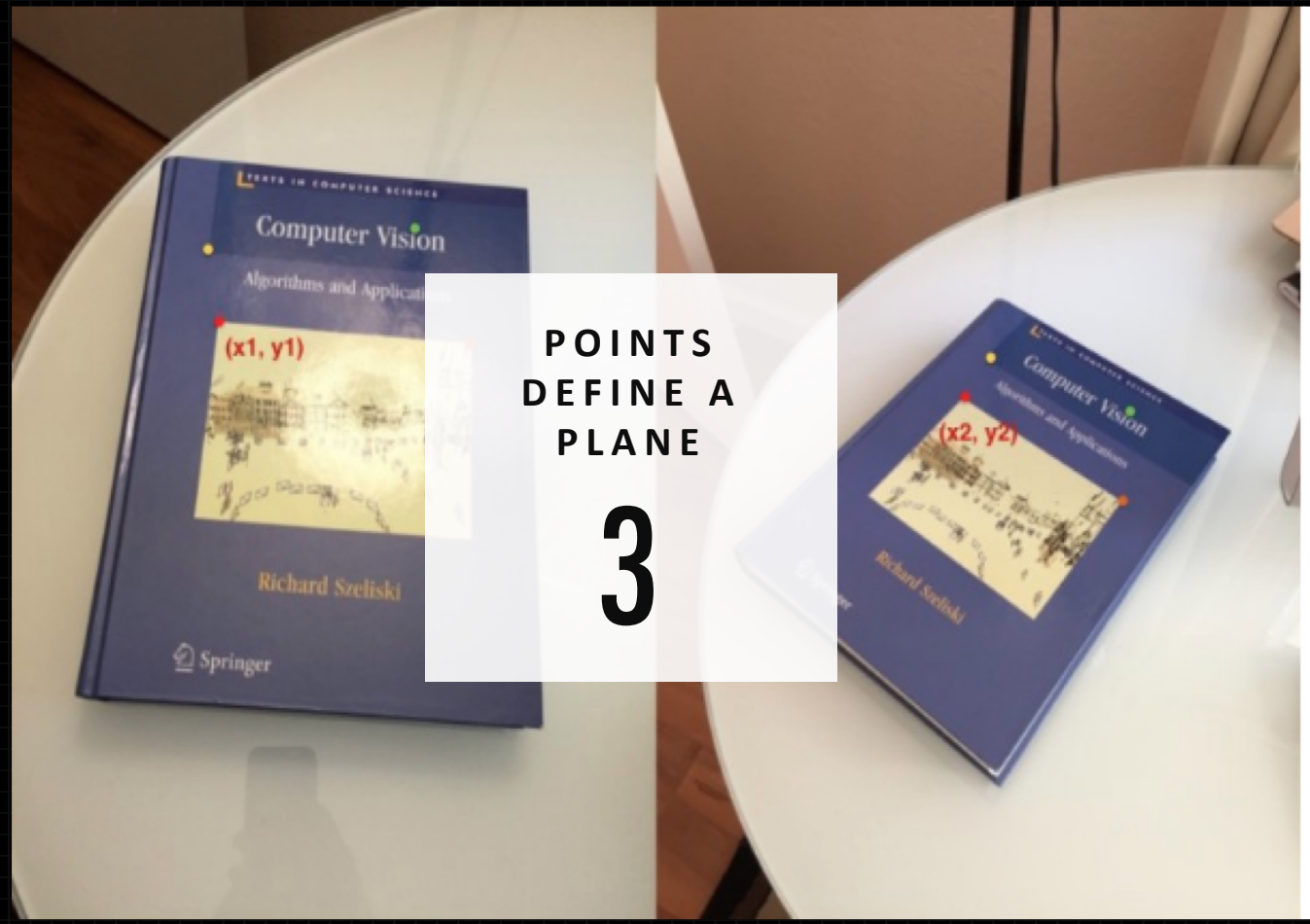




03 MEAN MASK ; YYC; XXC

04 REFERENCE IMAGE

PORTRAIT FCN+: PREPROCESSING

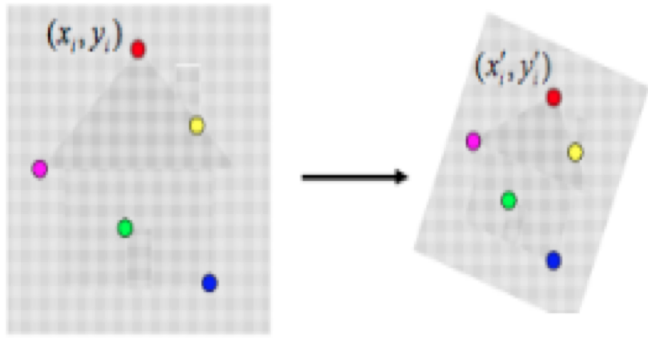


POINTS DEFINE A PLANE

3

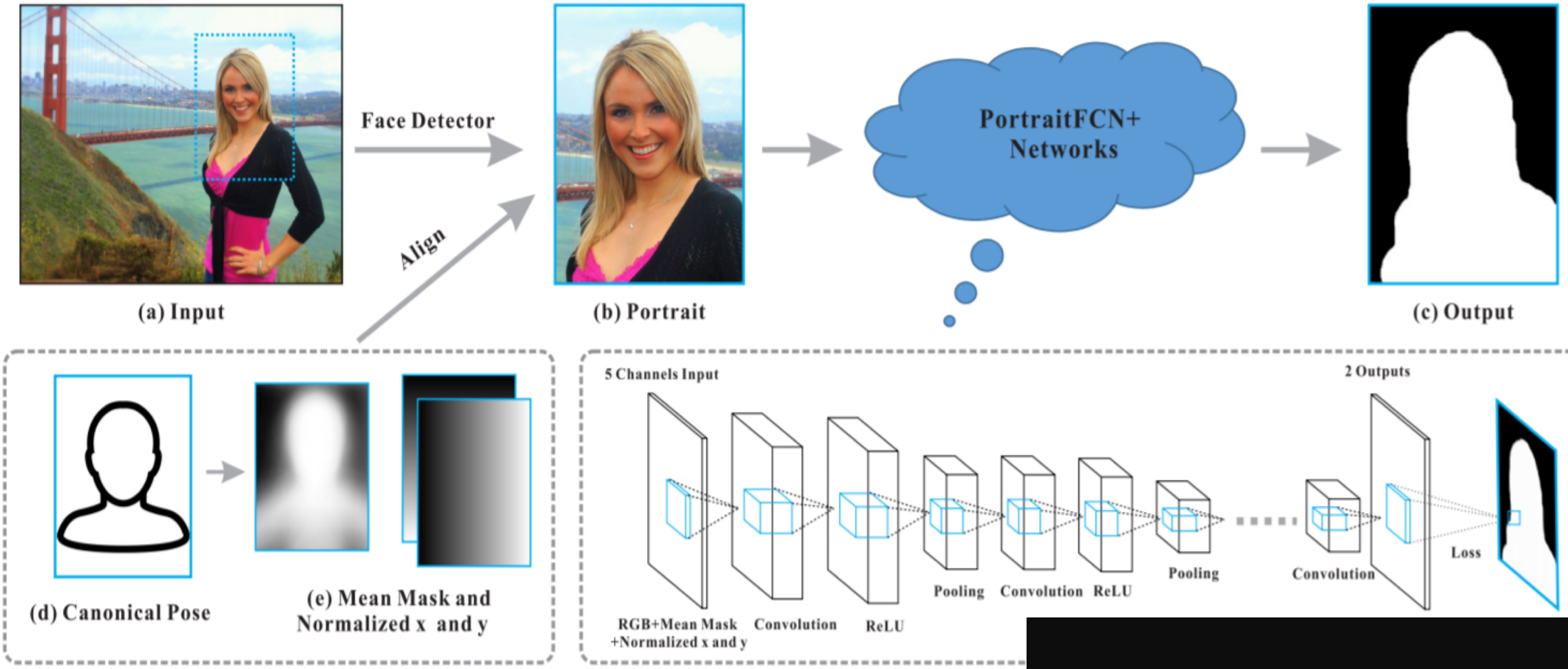
Fitting an affine transformation

- Assuming we know the correspondences, how do we get the transformation?



$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$\begin{bmatrix} \cdots & & & & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ \cdots & & & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}$$



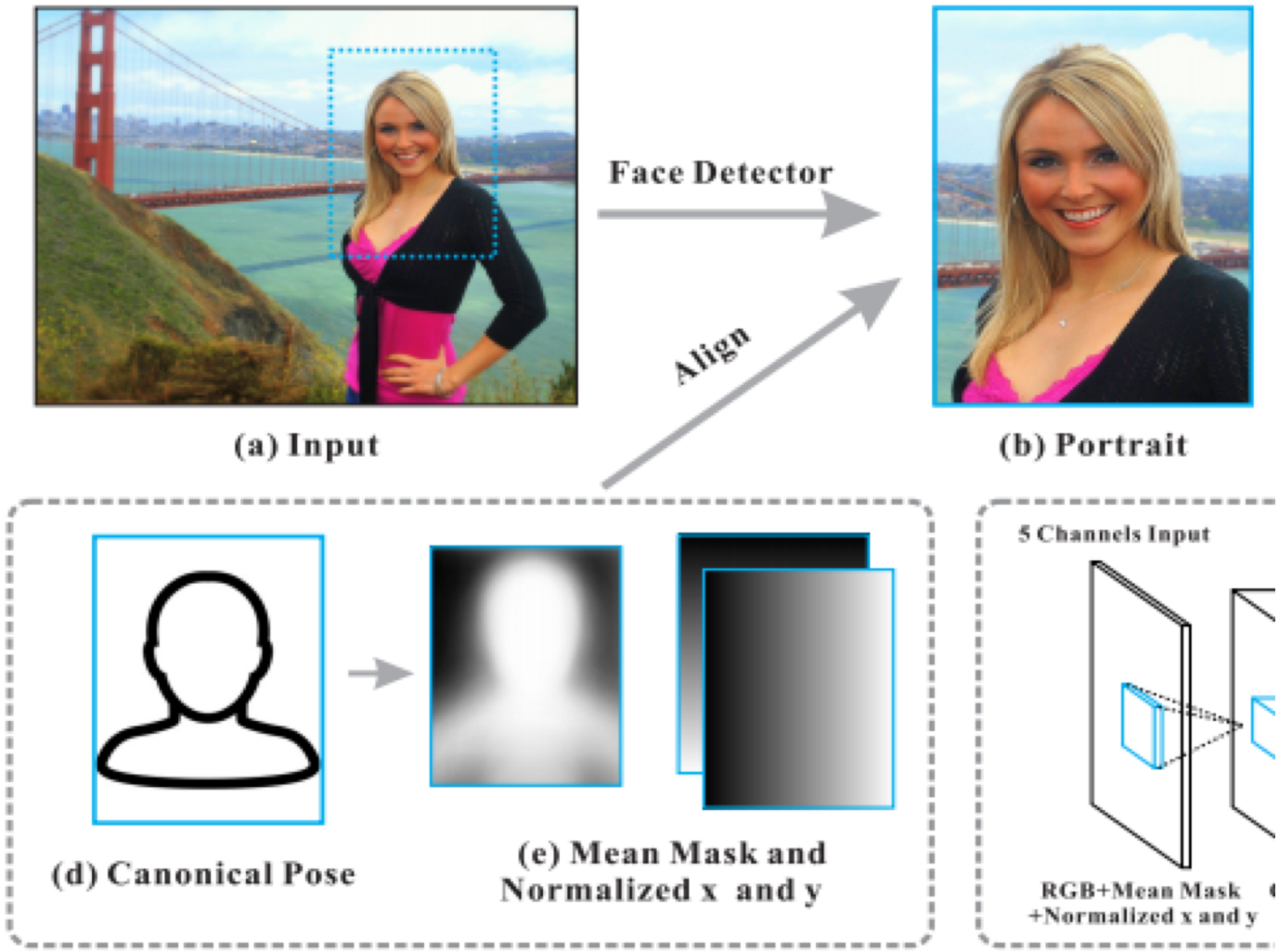
PORTRAIT FCN+

From Input to Trimap

PORTRAIT FCN+

Portrait-specific channels

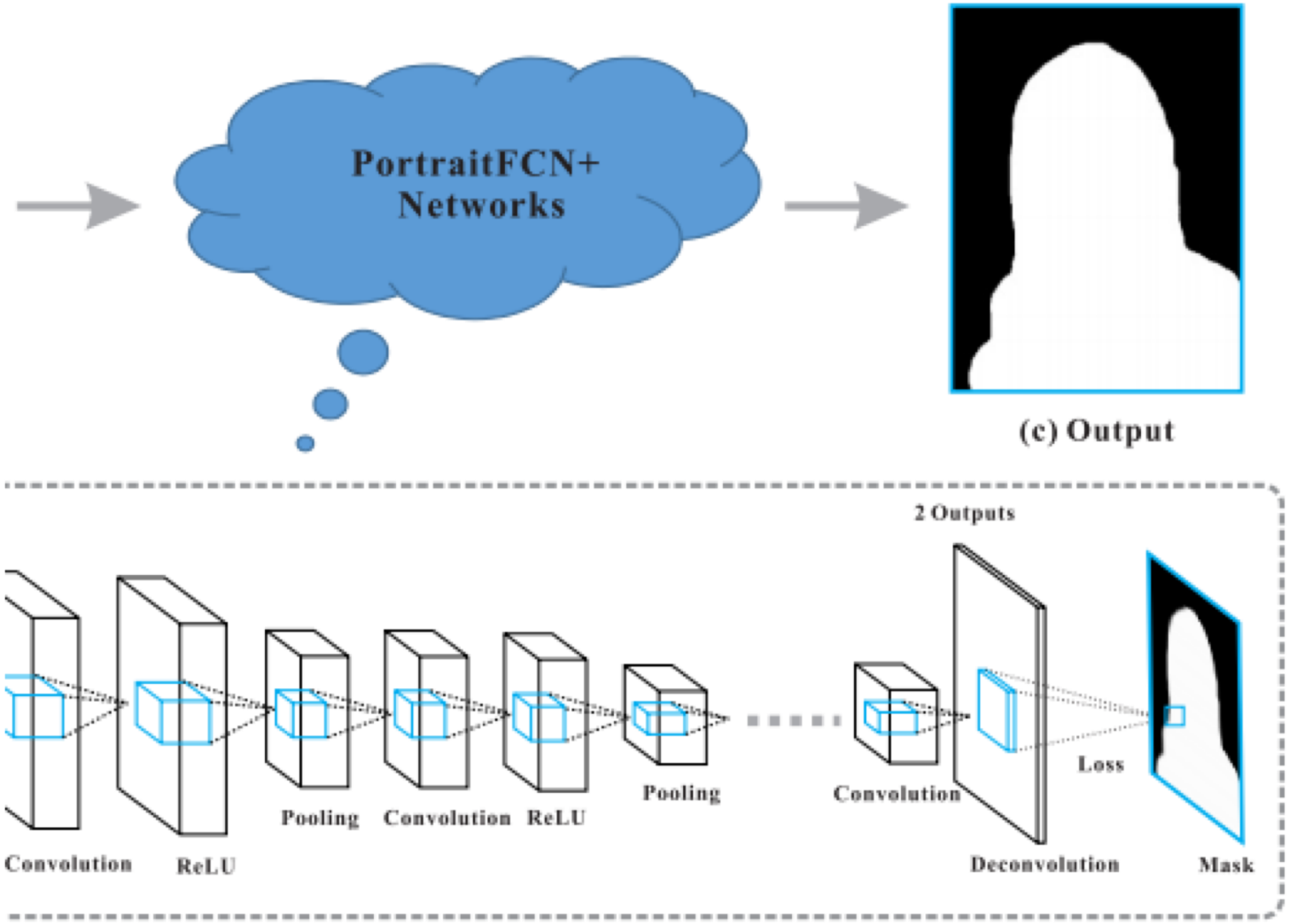
- Augment fully convolutional neural network (FCN) with portrait-specific channels
- Normalized x and y channels: homography transform between facial feature points and canonical pose
- Mean mask channel: subject-shaped region overlaid on actual subject as an estimate of final result

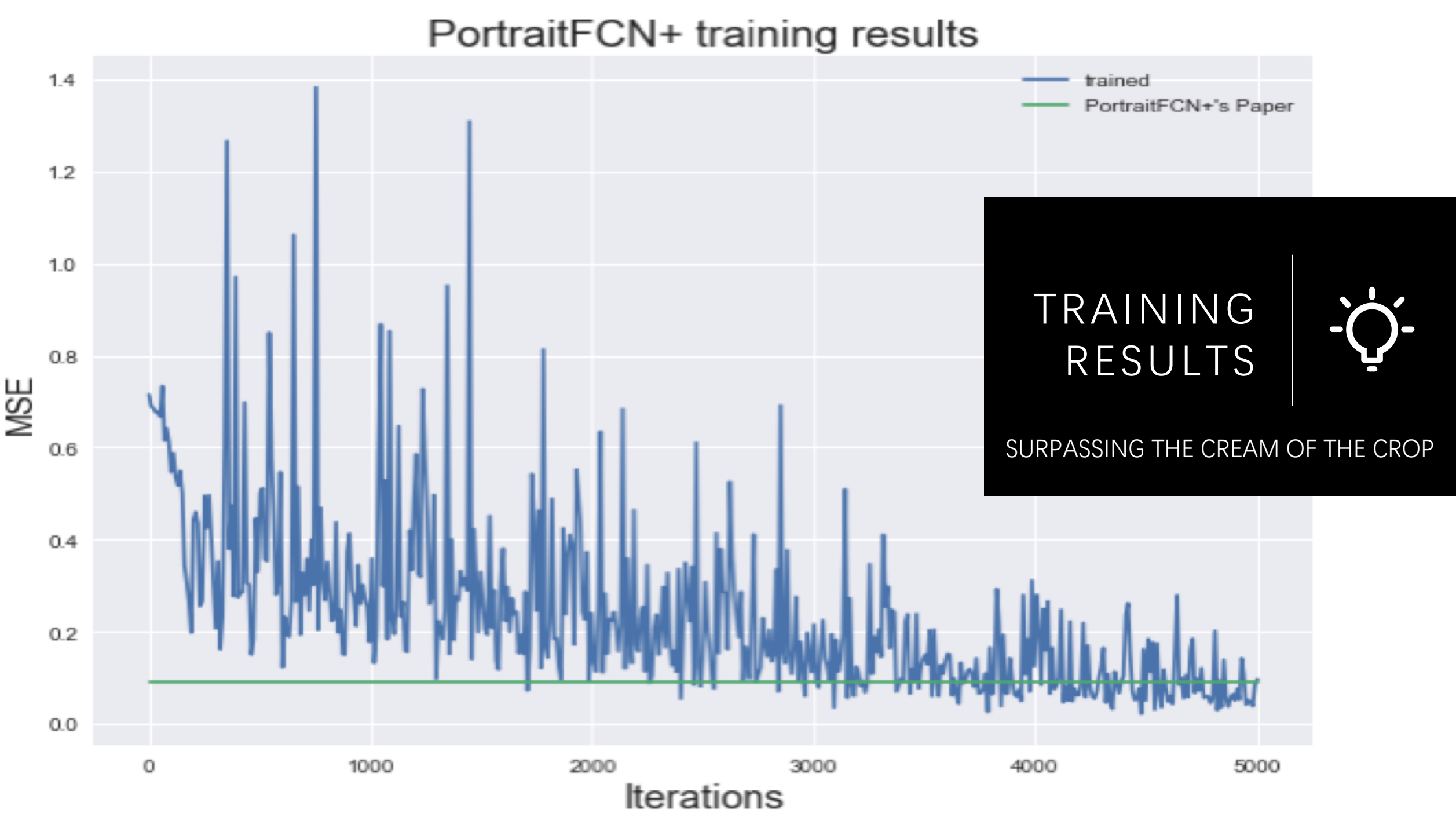


PORTRAIT FCN+

Layers of the neural network

- Convolution: extract image features, such as edges
- ReLU: nonlinear activation function
- Pooling: max or average of feature over a region
- Deconvolution: upsample previous layers to ensure output matches input size
- Loss: measure error between output and ground truth





TRAINING RESULTS

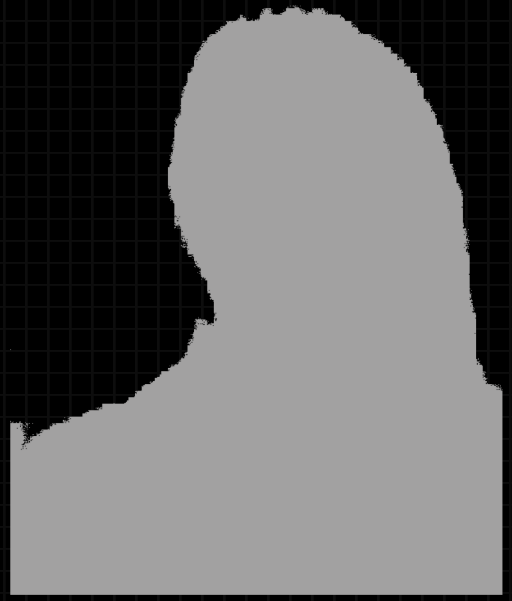
SURPASSING THE CREAM OF THE CROP

TRAINING RESULTS

Comparing to ground truth

01

ORIGINAL



02

PORTRAIT FCN+ MASK
OUR MEAN IOU: 93%

03

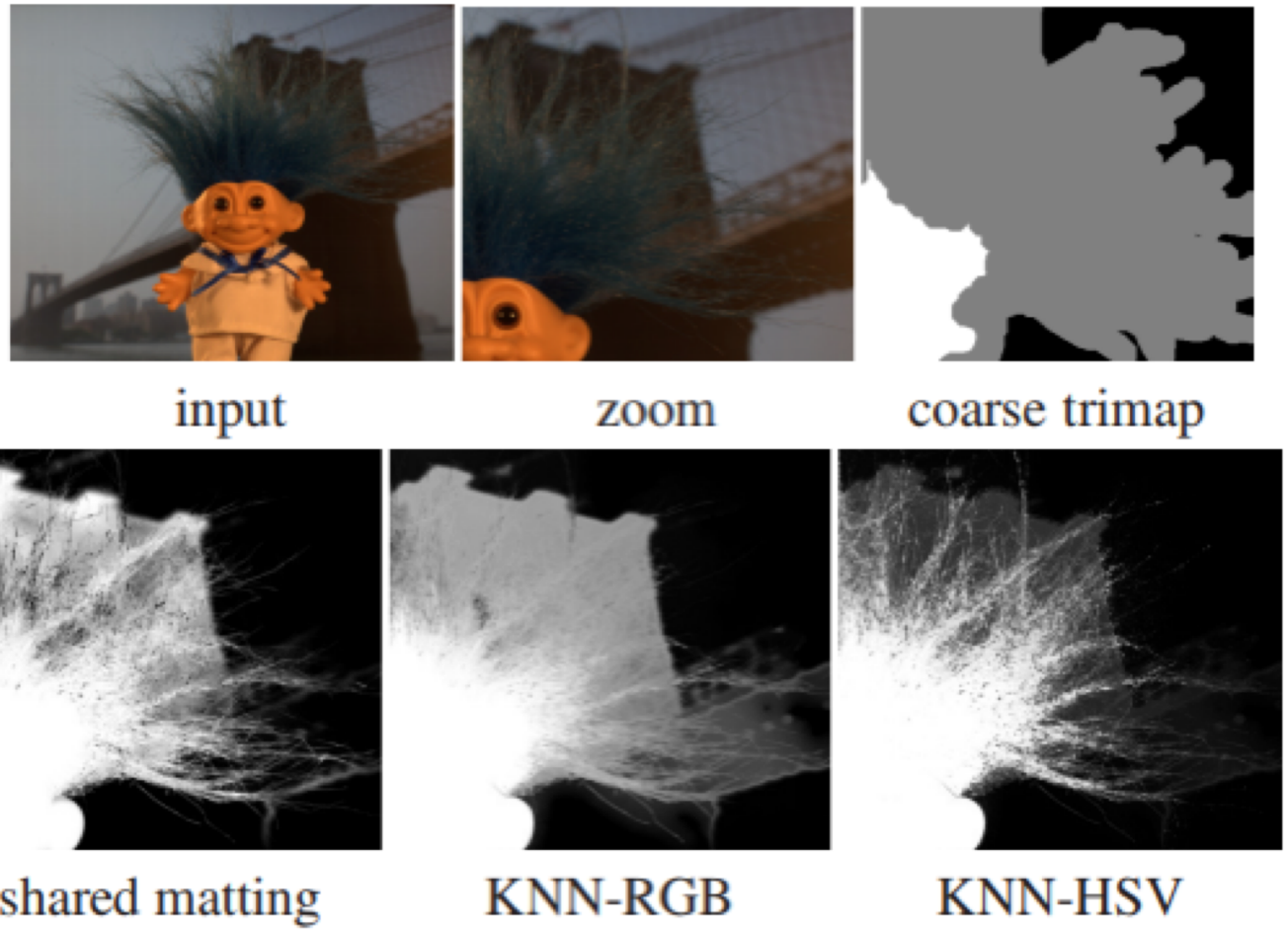
PAPER MEAN IOU: 95%

04

GROUND TRUTH

KNN MATTING

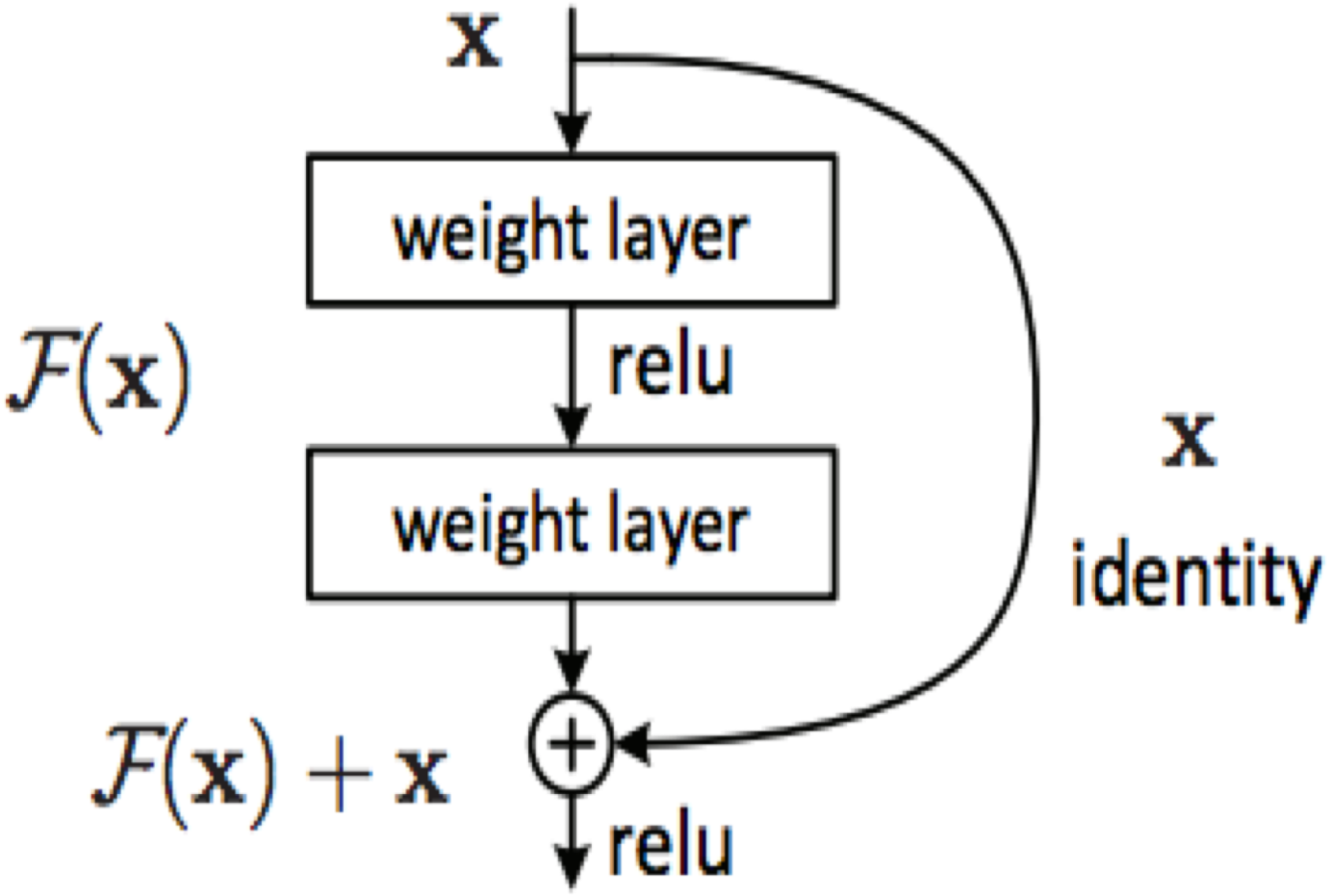
From Trimap to Alpha Matte



- We created trimaps by setting the unknown as 10 pixels away from segmentation boundary of Portrait FCN+
- We then applied it to KNN matting, which uses K nearest neighbours to get closed-form solution
- Closed-form solution utilizes the conjugate gradient method

RESIDUAL NETWORK (RESNET)

Traditionally Used to Avoid Vanishing Gradient, but we use to "refine" alpha matte



- Increased layer of network makes the earlier layers negligible (vanishing gradient)
- Adding the layers leads to high training error
- Imply an identity mapping of the input to create a direct path to the output; the latter layer learns only the residual

KNN MATTING & RESNET Results

From Trimap to Alpha Matte

1 Output of Portrait FCN+ (Mask)

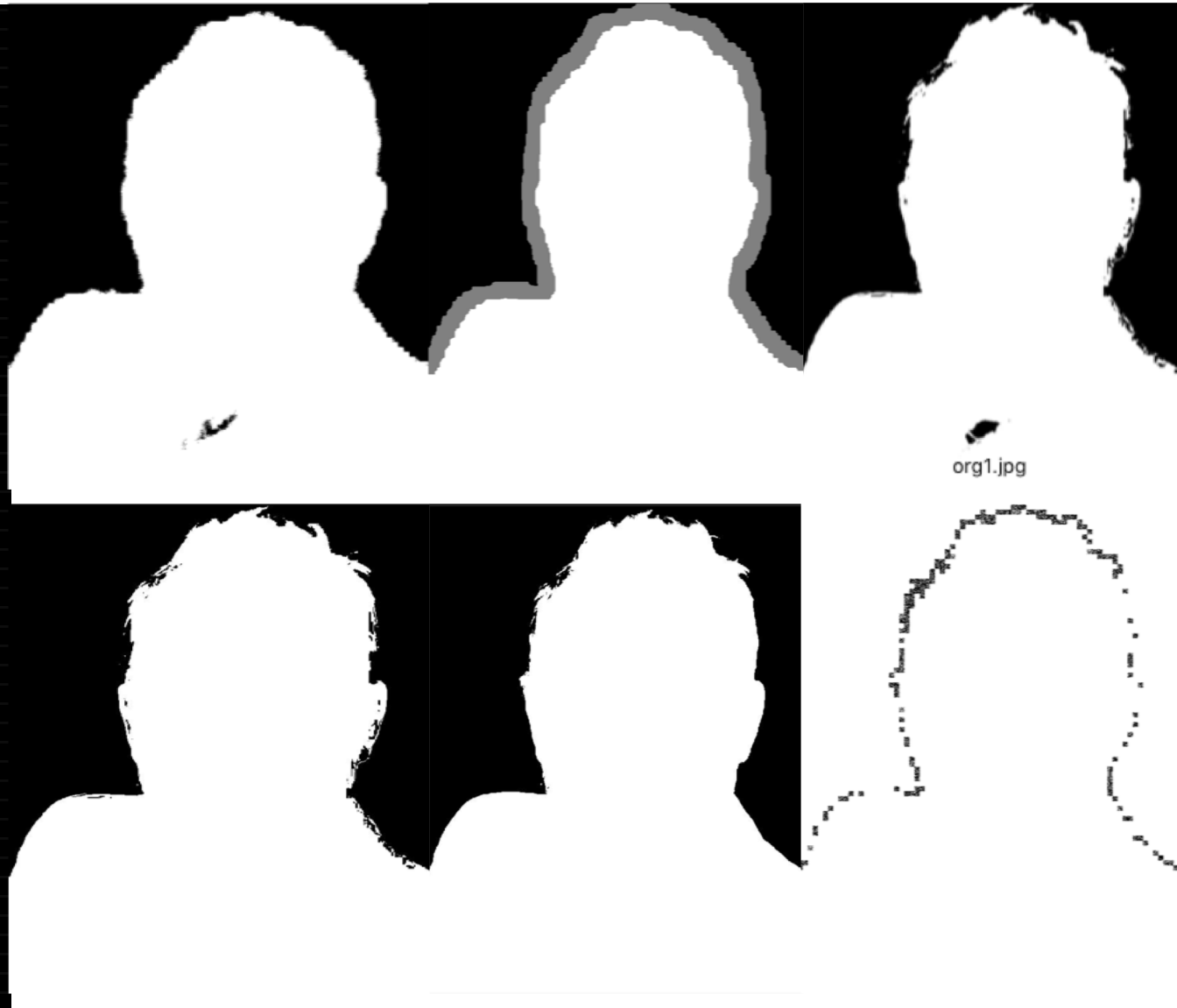
2 Trimap

3 KNN IoU: 95.9%

4 KNN+ Refinement (ResNet) IoU: 97.1%

5 Ground Truth

6 Difference Between 1 & 4



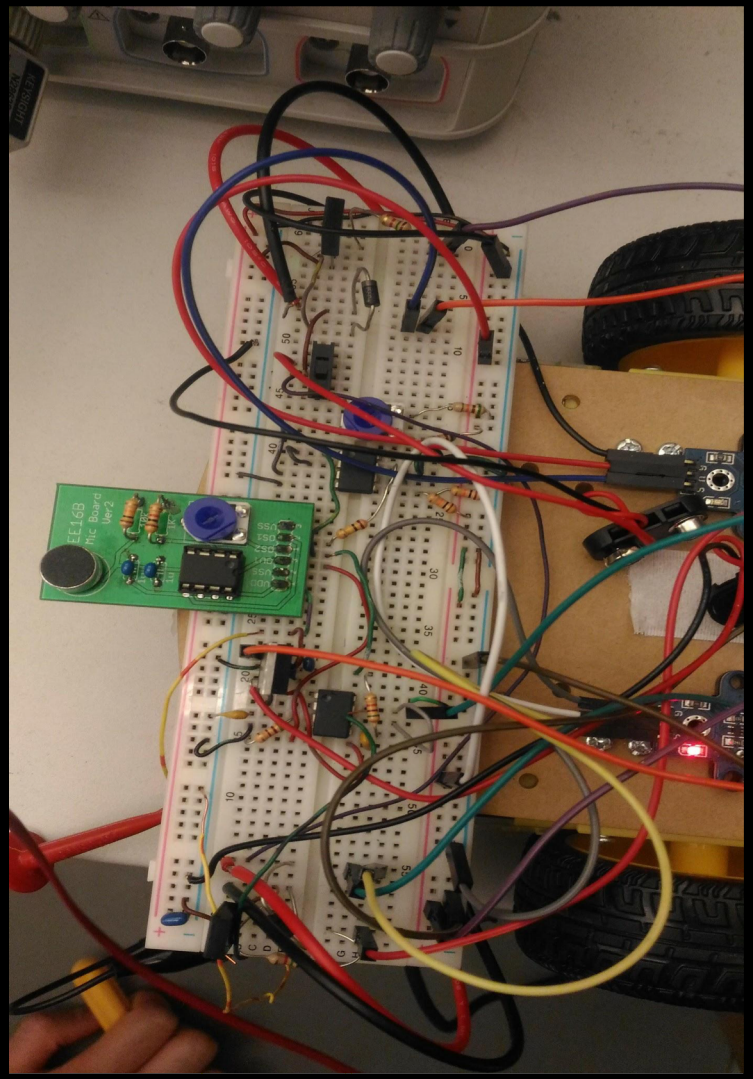
COMPARISON TO DIM (STATE OF THE ART)

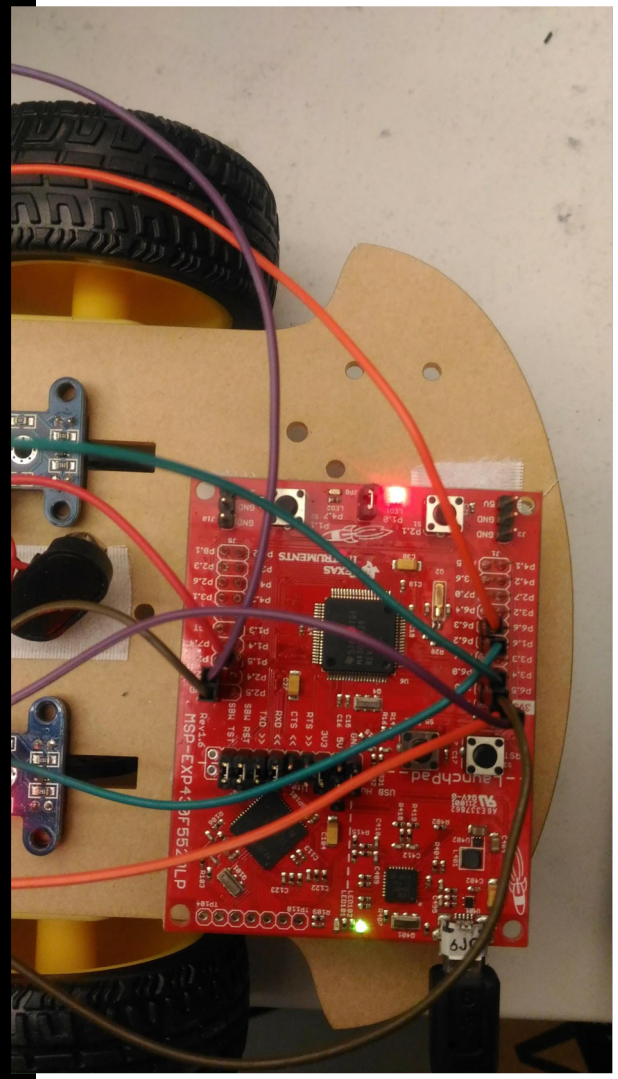
Ours achieved around 97% with KNN

Methods	[13]	[34]	[28]	[16]	[22]	[5]	[8]	Ours
Shared [13]	-	**60.0**	**78.5**	**79.6**	**69.7**	40.6	**57.8**	**83.7**
Learning [34]	40.0	-	**60.2**	**54.6**	**53.4**	27.3	35.1	**83.6**
Comprehensive [28]	21.5	39.8	-	25.8	43.3	20.4	29.2	**78.8**
Global [16]	20.4	45.4	**74.2**	-	**53.3**	30.0	42.0	**84.2**
Closed-Form [22]	30.3	46.6	**56.7**	46.7	-	25.0	38.1	**80.4**
KNN [5]	**59.4**	**72.7**	**79.6**	**70.0**	**75.0**	-	**73.3**	**97.0**
DCNN [8]	42.2	**64.9**	**70.8**	**58.0**	**61.9**	26.7	-	**83.7**
Ours	16.3	16.4	21.2	15.8	19.6	3.0	16.3	-

HARDWARE ADVANTAGE:

KNN-Matting





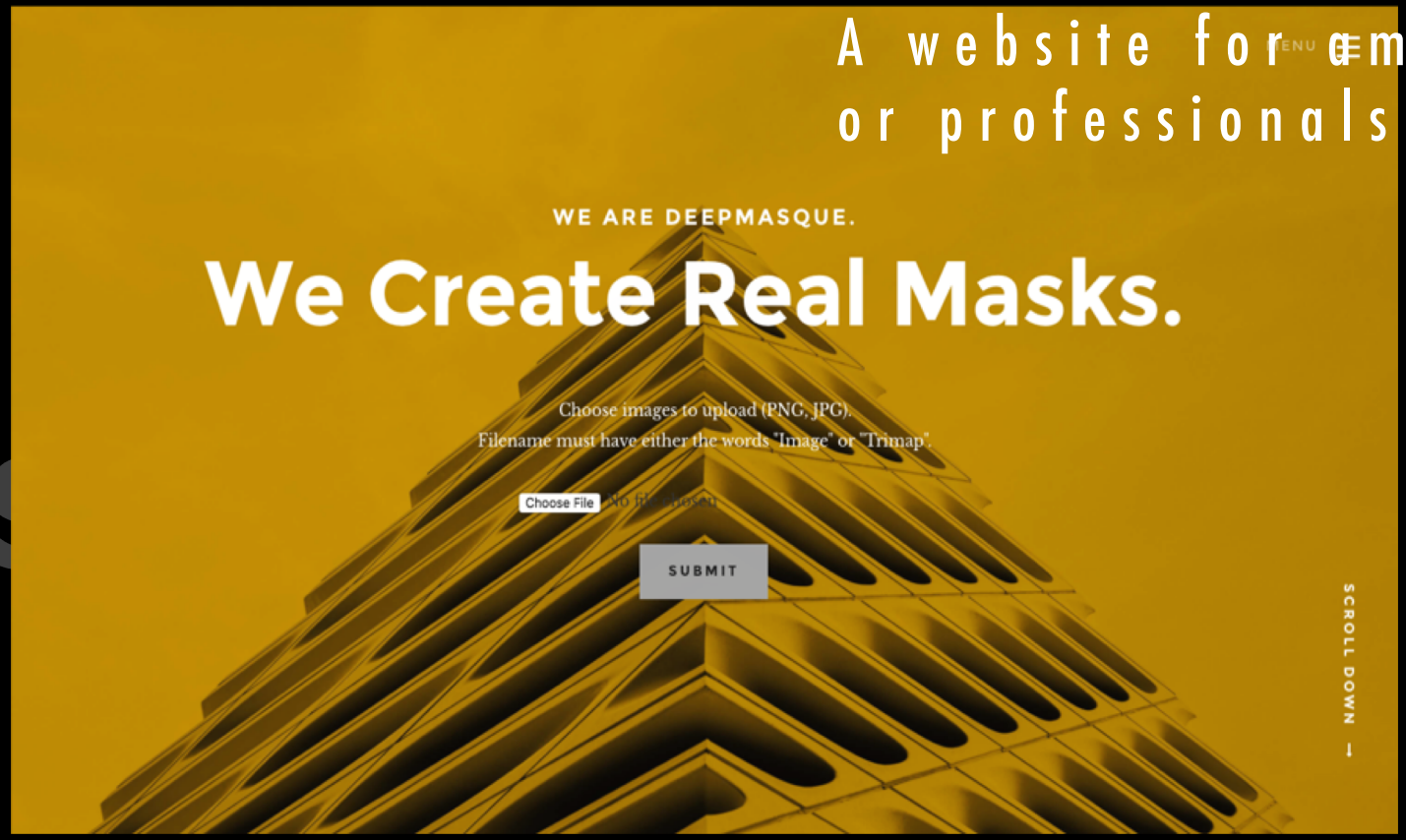
Launchpad: Signal Intake

Applied Signal of length 800 on the micboard which were then projected onto 2 PCA vectors and then applied with KNN as a linear system solver on the spot using a Energia Arduino program. The length 800 voice input represents a column of image which PCA will be applied on.

Significance: We prove that the masking interface for the KNN-matting section is capable of operating on a hardware of only 4 KB, virtually nothing by modern standards. This means we essentially have completely wiped out the original state of the art dependency on memory that DIM took up!!

USER INTERFACE

A website for amateurs or professionals

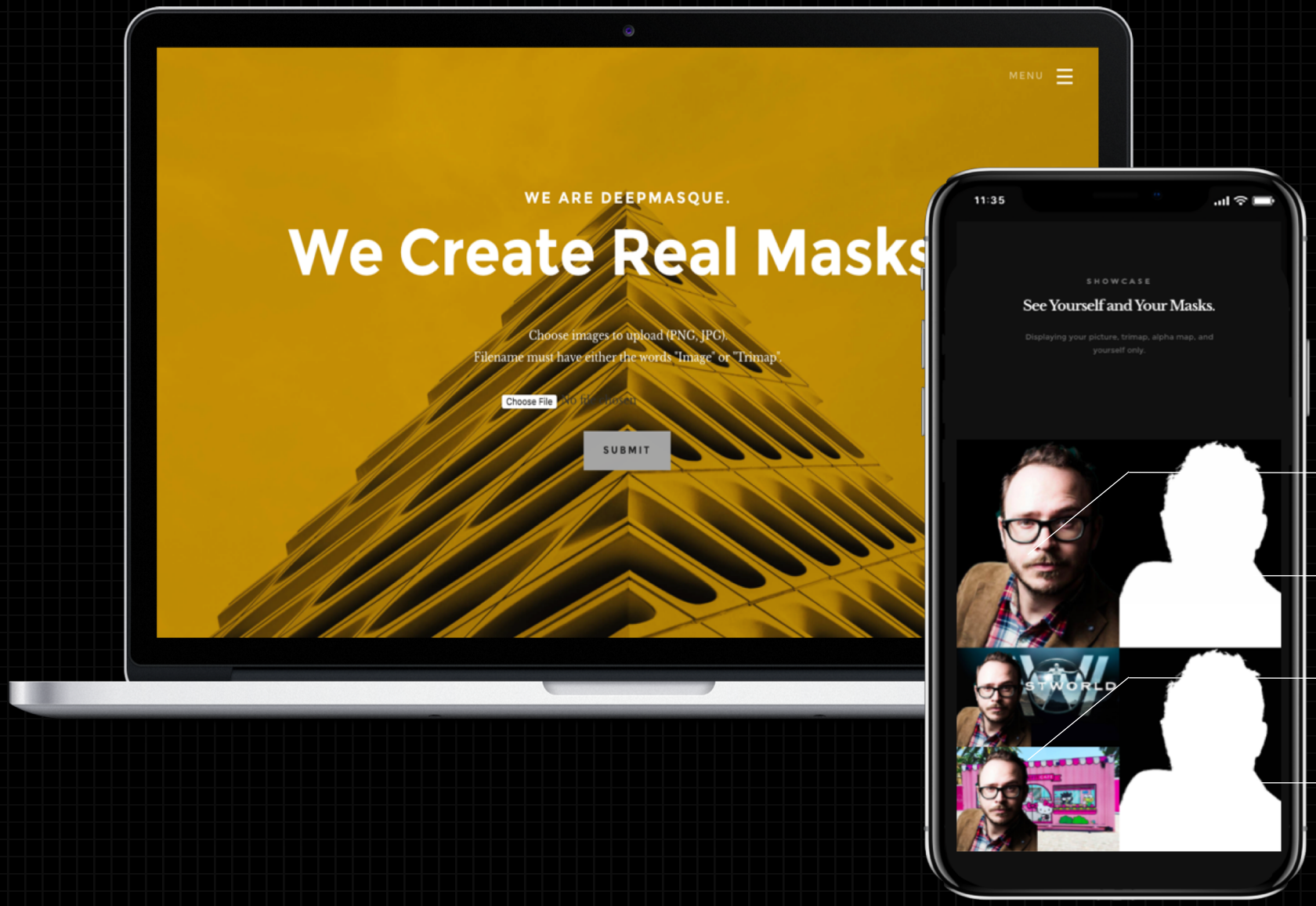


Our website enables the user to upload their picture and submit it to obtain two masks, the trimap and the alpha matte, and a final representation of the subject on a different background.

USER INTERFACE

Laptop & Cellphone End

"Our website enables the user to upload their picture and submit it to obtain two masks, the trimap and the alpha matte, and a final representation of the subject on a different background."



01. Original Input Image

02. Trimap

03. Final Products

04. Alpha Matte

Low-Tech Demo Goals

- ~~Gain Familiarity with TensorFlow Framework~~
- ~~Develop proficiency in neural networks, CNNs, image segmentation, and FCNs~~
- ~~Understanding two Arxiv papers: 1) Portrait FCN+ 2) Deep Matted Learning Paper~~
- ~~Training datasets for PFCN+ via GPUs (cudas) or AWS (clusters)~~
- ~~Develop Potrait FCN+ and Deep Matted Learning model in Tensorflow~~
- ~~Transform & Present the framework in a user input-allowed format~~

… And Beyond

- Use KNN matting to create alpha matte from trimap
- Create user interface in web and mobile development that produces masking images using our model
- Prove the memory reduction of our model compared to planned model with primitive hardware device via hardware demo
    - Achieved similar IoU to DIM with less costly algorithm, opening a potential renew research into KNN-Matting
        - Kept the whole runtime/process time under ~5 minutes (due to KNN unoptimized)
            - Currently working on 3-D Modeling

LEARNING PATHS

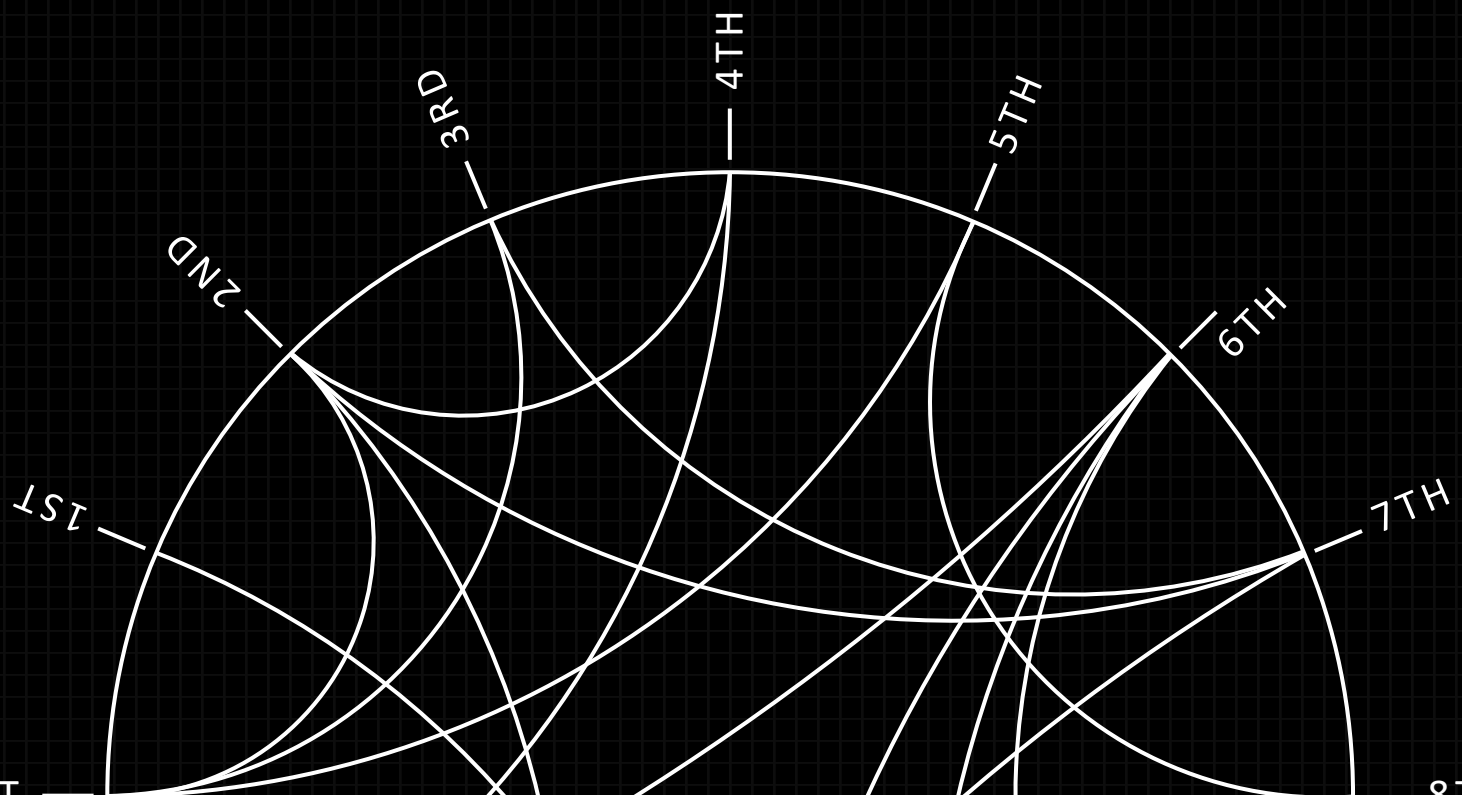
+++

+++++

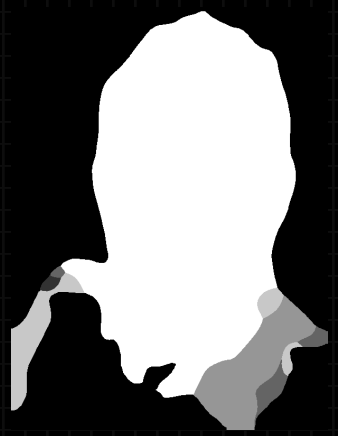
WE STARTED WITH REPRODUCING XIAOYONG SHEN'S PAPER (IN MATLAB) BUT IN TENSORFLOW.

- 0TH POINT: LEARNING TENSOR-FLOW AND DEEP LEARNING/COMPUTER VISION FOR THE FIRST TIME
  - WE FIRST DID A FCN-8S (USED IN SEGMENTATION FOR SELF-DRIVING CAR) TO SEE HOW THAT WOULD LOOK LIKE. THE LOW IOU WAS ~55%

- 1ST ITERATION - WE THEN IMPLEMENTED PORTRAITFCN+ WHICH HAD MULTIPLE ISSUES:
  - 1) WE DIDN'T SPECIFY PADDING (SO TENSORFLOW JUST MADE THE INCORRECT PADDING FOR US)
  - 2) BUGS WHICH WE FIXED BY USING TF.LAYERS LIBRARY INSTEAD OF TF.NN
  - 3) INCORRECT LOSS FUNCTION AND
  - 4) FAILED TO RUN CAFFE IN TENSORFLOW (WHERE IN PAPER, XIAOYONG & HER TEAM USED CAFFE IN MATLAB) (ACCURACY: 75% IOU)



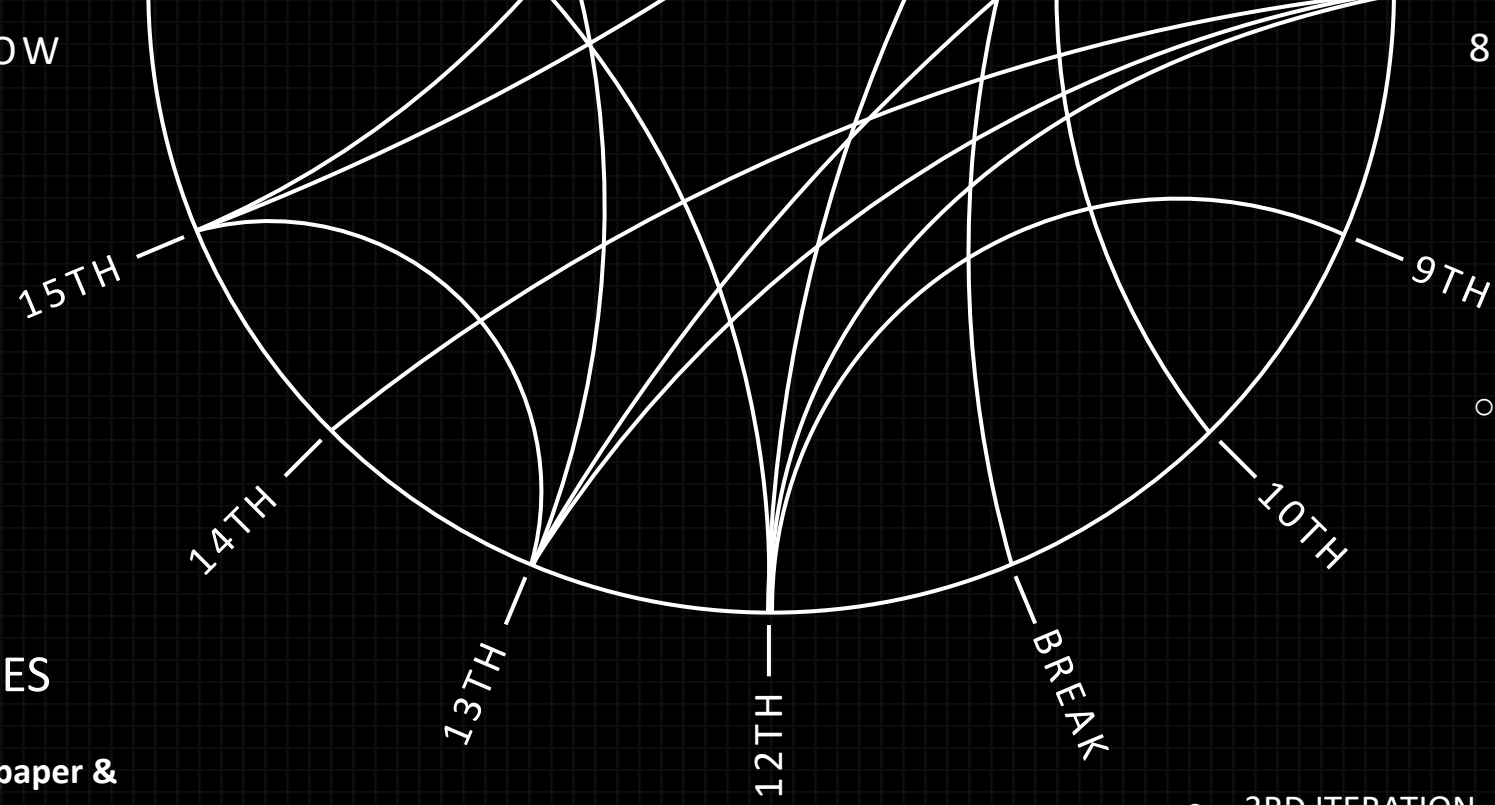
WE TRIED TRAINING LOCALLY AND THAT FROZE THE LAPTOP, SO WE TRAINED ON 8 CPUS/2 GPUS ON AWS.



01

LEARNING OUTCOMES

- **Analyzing research paper & realizing and validating the implementation of paper**
- **Learned how to use/setup AWS**
- **Learned encoder-decoder network, Resnet, fcn-8s**
- **Computer vision techniques, transforms, Portrait FCN+, KNN application to masking**
- **Basic web development**
- **Facial feature extraction & image rotations**
- **Importance of reproducibility**

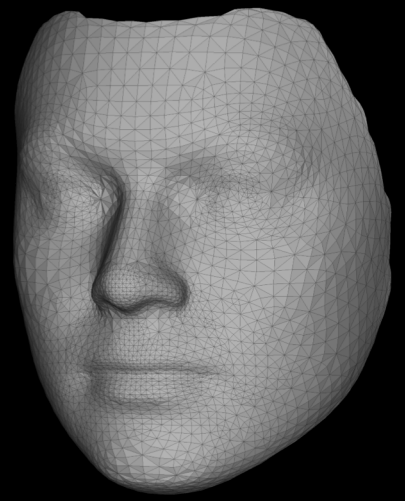


- 2ND ITERATION - FIXED ALL THE BUGS AND INITIALIZE THE CONV-NET PORTION OF PORTRAIT FCN+ WITH VGG (ACCURACY 92%)
- 3RD ITERATION - WE TRAINED THE CONV-NET WITHOUT INITIALIZATION, WHICH DIDN'T CONVERGE (ACCURACY 70% IOU)
  - WE RECEIVE A REPLY FROM THE DIM PEOPLE THAT THEIR DATA CANNOT BE USED FOR ANY COMMERCIAL PURPOSES, WHICH COMPLICATED OUR MENTORSHIP WITH TWINDOM.
- 4TH ITERATION - WE REALIZED THAT PORTRAIT FCN+ MIGHT WORK WELL WITH KNN-MATTING. IMPLEMENTED THE KNN-MATTING.
- 5TH ITERATION - ADDED RESNET AND SOME POST-PROCESSING TO IMPROVE IOU (96% IOU)
- IMPLEMENTED ***TWO*** ITERATIONS OF UI

3D Morphable Face Model

Isomap texture extraction

- to obtain a pose-invariant representation of the face texture
- Linear scaled orthographic projection camera pose estimation
- Linear shape-to-landmarks fitting
- Expression fitting, and 6 linear expression blendshapes
- Edge-fitting, heavily inspired



Surrey Morphable Face Model

Surrey Face Model is a PCA model of shape variation built from 3D face scans. It comes with uv-coordinates to perform texture remapping

WE ARE DEEPMASQUE

THANK YOU

Questions?

https://github.com/leoli3024/Portrait-FCN-and-3D-reconstruction