

# DeFINe – Deep Facial Inpainting Network

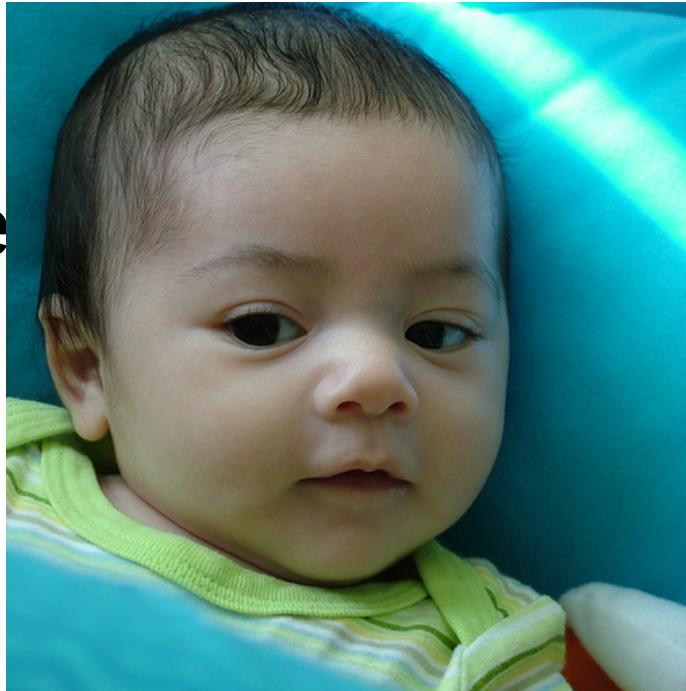
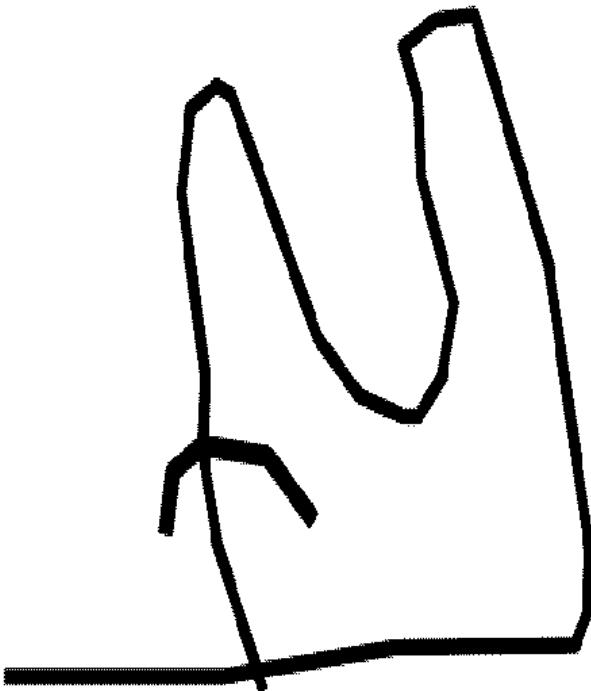
- Kevin Gellhaus
- Marcel Früh
- Micha Schilling

# Recap - Motivation

- Durchführung eines Projektes mit nahezu unbegrenzten Daten
- Verbesserung existierender Inpainting Lösungen in Bezug auf Gesichtsrekonstruktion
- Einfaches Retuschieren von z.B Passfotos, Porträts

# Daten

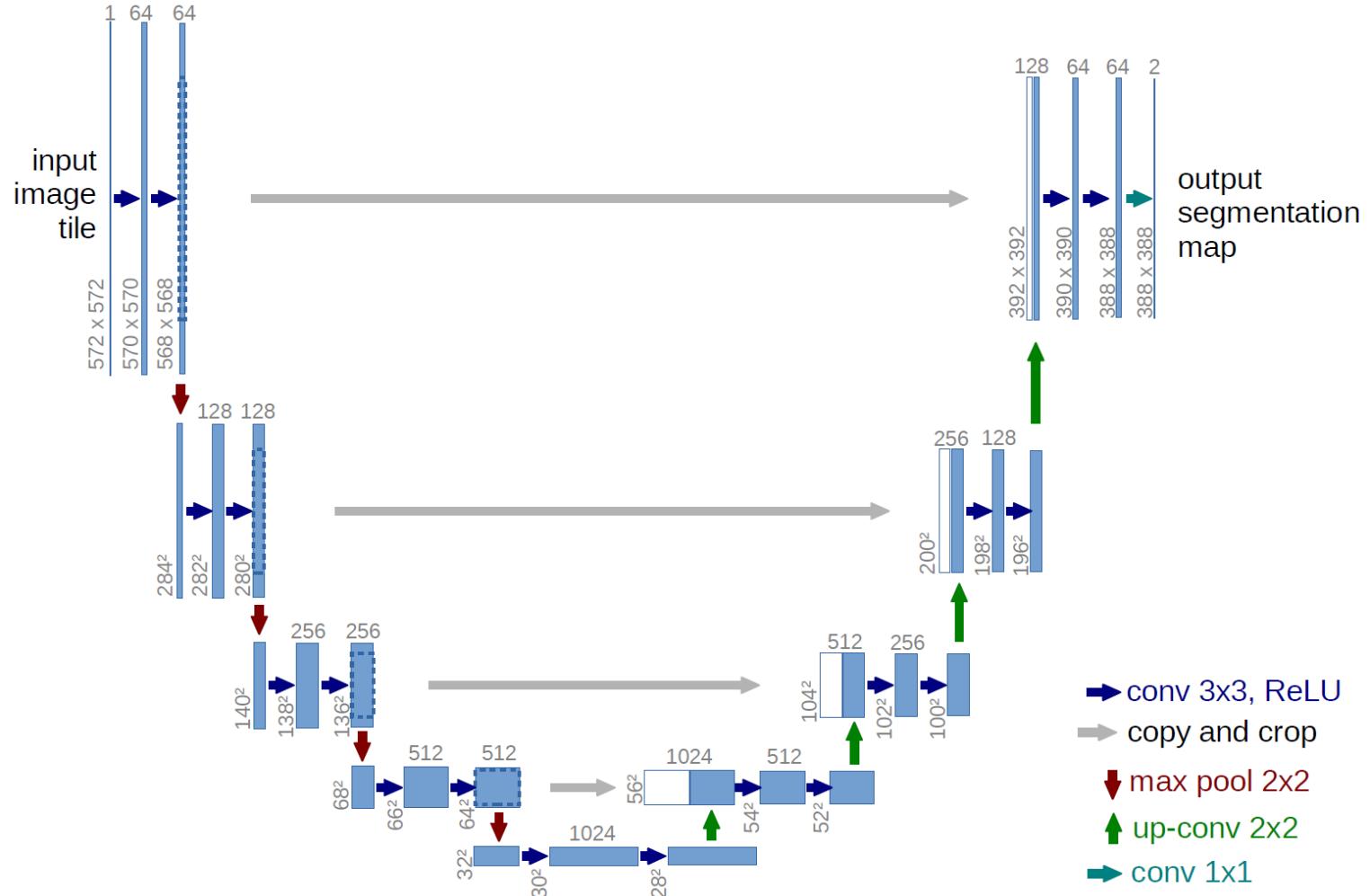
- Trainingsdaten: Flickr-Faces-HQ Dataset (FFHQ)
- Masken: QD-IMD: Quick Draw Irregular Mask Dataset
- 70.000 Gesichter und 50.000 Masken
- Auflösung: 512x512



# Model

- U-Net Architektur (Convolutional Networks for Biomedical Image Segmentation (Ronnenberger et al.)

# Mod • U-N Ima



# Model

- U-Net Architektur: “Convolutional Networks for Biomedical Image Segmentation” (Ronneberger et al.)
- Upsampling statt Transposed Convolution
- Stride statt Max-Pooling
- Partial Convolutions: “Image Inpainting for Irregular Holes Using Partial Convolutions” (Liu et al.)

# Partial Convolution

$$x' = \begin{cases} W^T(X \odot M) \frac{\text{sum}(1)}{\text{sum}(M)} + b & \text{falls } \text{sum}(M) > 0 \\ 0 & \text{sonst} \end{cases}$$

- $X \rightarrow$  Bild oder Ausgabe vom vorherigen Layer
- $W \rightarrow$  Gewichte der Conv. Filter
- $M \rightarrow$  Zu  $X$  gehörende binäre Maske (0: maskiert, 1: originales Bild)
- $1 \rightarrow$  Einser-Matrix mit Größe des aktuellen Filters
- $b \rightarrow$  Bias der Conv. Filter
- $\odot \rightarrow$  Elementweise Multiplikation

# Partial Convolution – Anpassung der Maske

$$m' = \begin{cases} 1 & \text{falls } \text{sum}(M) > 0 \\ 0 & \text{sonst} \end{cases}$$

Zielstellung:  
Einfluss von verschiedenen Lossfunktionen

# Lossfunktionen - V0

- Einfacher L1 Loss
- $L = \frac{1}{N} ||O - GT||_1$

# Lossfunktionen - V1

- $L_{hole} = \frac{1}{N} \|(1 - M) \odot (O - GT)\|_1$
- $L_{valid} = \frac{1}{N} \|(M) \odot (O - GT)\|_1$

# Lossfunktionen - V2

- $L_{hole} = \frac{1}{N} \|(1 - M) \odot (O - GT)\|_1$
- $L_{valid} = \frac{1}{N} \|(M) \odot (O - GT)\|_1$
- $L_{perceptual} = \sum_{p=0}^{P-1} \frac{\|\Psi_p^O - \Psi_p^{GT}\|_1}{N} + \sum_{p=0}^{P-1} \frac{\|\Psi_p^C - \Psi_p^{GT}\|_1}{N}$

# Perceptual Loss – Was ist das?

- $$L_{perceptual} = \sum_{p=0}^{P-1} \frac{||\Psi_p^O - \Psi_p^{GT}||_1}{N} + \sum_{p=0}^{P-1} \frac{||\Psi_p^C - \Psi_p^{GT}||_1}{N}$$
- $\Psi$  ist die Ausgabe des p-ten Layers eines vortrainierten VGG16 Netzes
  - Max-Pool 1, Max-Pool 2, Max-Pool 3
- “Controlling Perceptual Factors in Neural Style Transfer” (Gatys et al in Tübingen!)
- C → Vorhersage, bei welcher die nicht-maskierten Pixel auf den Ground Truth gesetzt werden

# Lossfunktionen - V3

- Noch 2 bzw. 3 weitere Lossfunktionen
- Vorstellung im Abschlussvortrag

# Training

- TCML Cluster
- Batch Size: 16 bzw. 6
- BatchNormalization
- Learning Rate: 2e-4
- 3 Tage

# Vergleich: V0

Image with Mask



Prediction



# Vergleich: V1

Image with Mask

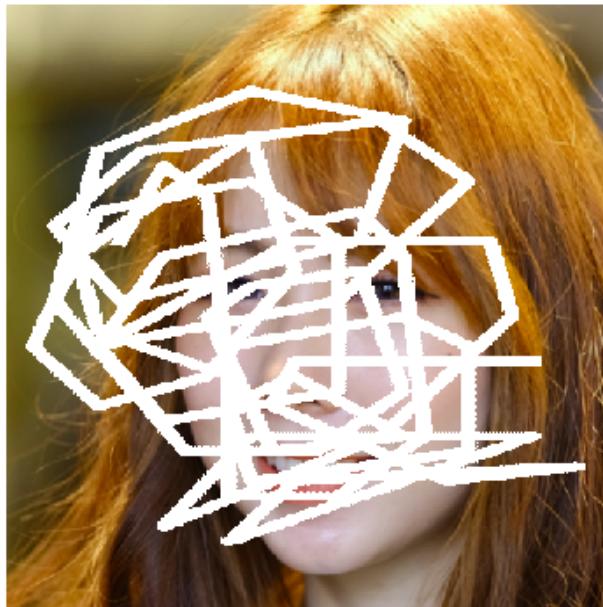


Prediction



# Vergleich: V2

Image with Mask



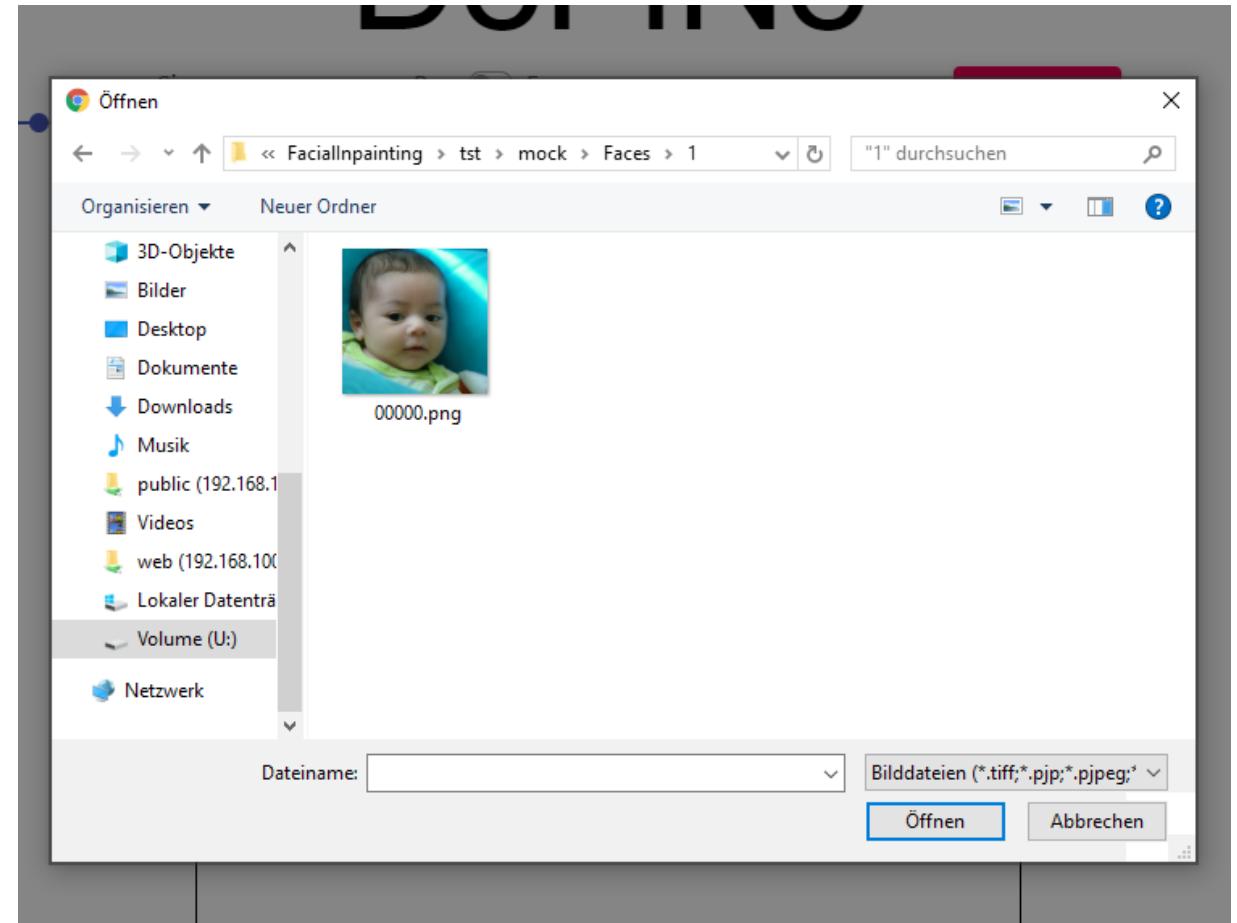
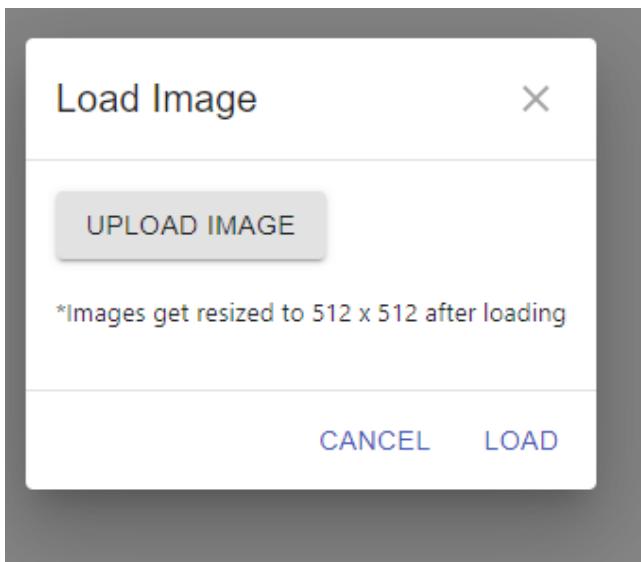
Prediction



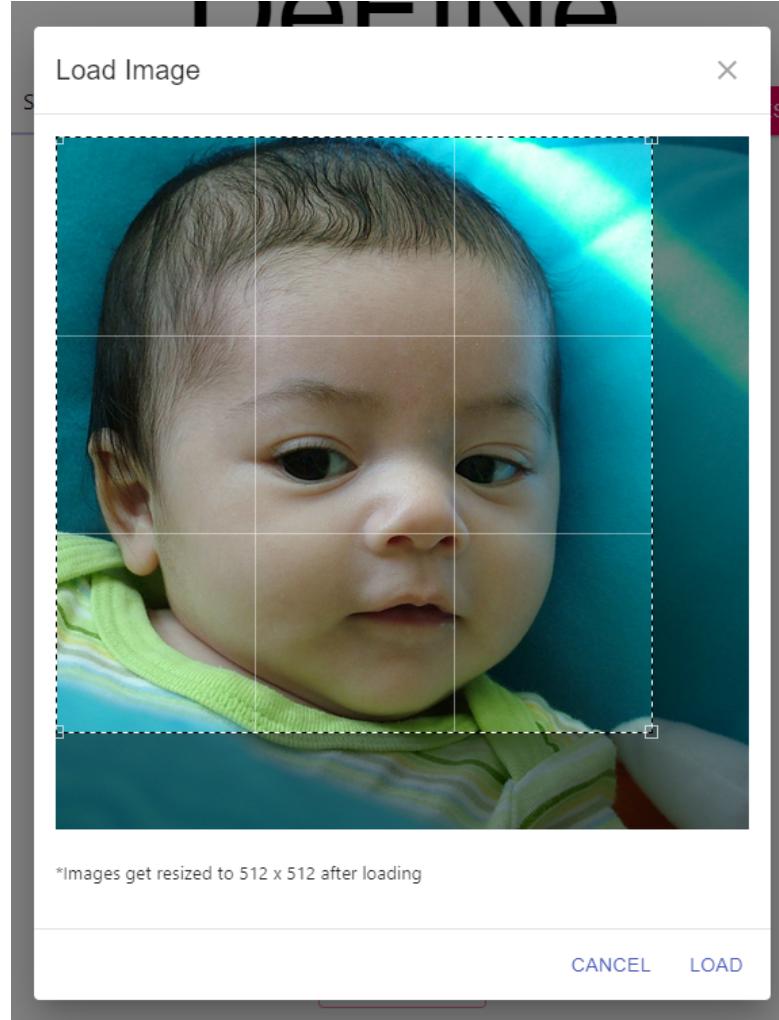
# GUI



# GUI



# GUI



# GUI



# GUI



```
[[[1,134,148,0],[1,134,148,0],[0,130,144,0],[0,127,141,0],[0,126,140,0],[0,128,143,0],[0,125,140,0],[0,121,136,0],[0,122,140,0],[0,122,152,0],[2,130,149,0],[0,127,146,0],[2,126,145,0],[2,125,142,0],[1,127,146,0],[0,125,144,0],[0,125,143,0],[0,124,144,0],[0,126,147,0],[0,120,138,255],[0,118,136,255],[0,115,134,255],[0,117,136,255],[0,144,255],[0,123,142,255],[0,121,141,255],[0,121,142,255],[0,121,143,255],[0,124,145,255],[0,123,144,255],[0,123,142,255],[0,122,141,255],[0,125,255],[0,120,135,255],[0,116,132,255],[0,115,131,255],[0,116,131,255],[0,116,129,255],[0,114,128,255],[0,111,124,255],[0,108,122,255],[0,100,255],[0,111,129,255],[0,113,131,255],[1,117,133,255],[1,119,135,255],[1,117,135,255],[0,120,138,255],[0,120,137,255],[0,117,134,255],[0,114,255],[0,114,132,255],[1,113,131,255],[0,108,126,255],[0,105,123,255],[0,106,123,255],[0,105,123,255],[0,104,123,255],[0,105,125,255],[0,100,255],[0,115,129,255],[0,115,128,255],[0,114,126,255],[0,113,127,255],[114,133,117],[0,110,128,0],[0,110,127,0],[0,115,131,0],[0,119,134,0],[0,108,128,0],[0,106,122,0],[0,108,121,0],[0,108,124,0],[1,110,129,0],[0,107,126,0],[0,108,126,0],[0,109,128,0],[0,109,128,0],[0,107,125,0],[1,100,121,0],[2,98,116,0],[0,94,110,0],[1,92,111,0],[2,97,117,0]]
```

# Ausblick

- Implementierung von restlichen Lossfunktionen
- API um GUI und Netzwerk zu verbinden
- GUI styling

# Literaturverzeichnis

- <https://github.com/NVlabs/ffhq-dataset>
- <https://github.com/karfly/qd-imd>
- <https://arxiv.org/abs/1505.04597> U-Net

# Vielen Dank

