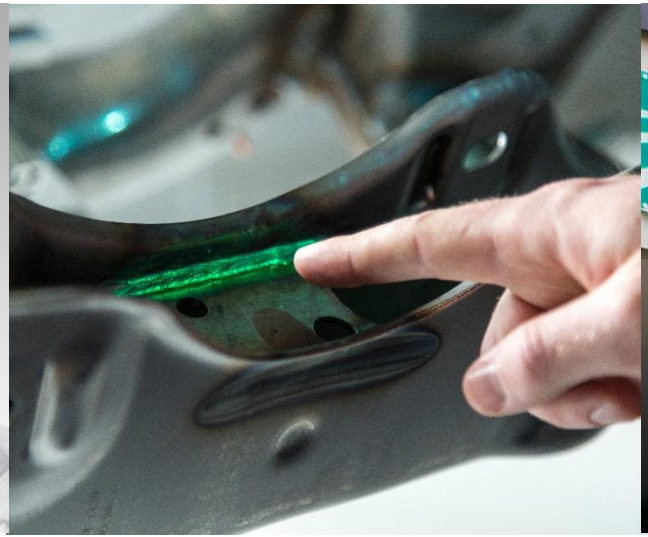
TRUSTWORTHY AI (for Secure Authentication)

VISUM SUMMER SCHOOL

Peter Eisert

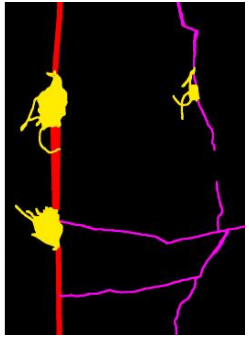
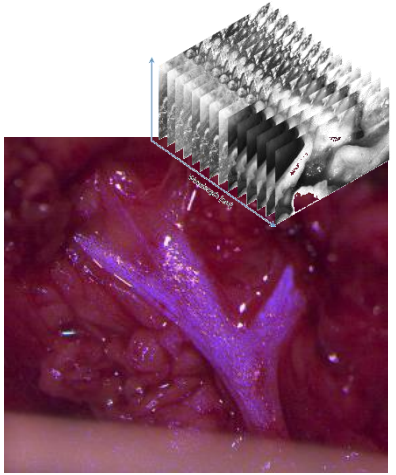


About Me

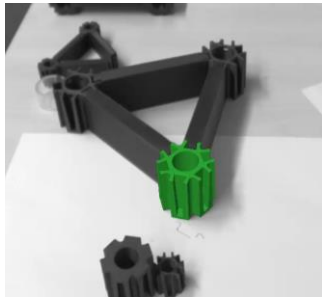
- PhD University Erlangen, Facial Expression Analysis
- Research Fellow, ISL Stanford
- Computer Vision & Graphics Group, Fraunhofer HHI, Berlin
- Professor Visual Computing, Humboldt University, Berlin
- Head of Vision & Imaging Department, Fraunhofer HHI

Research Areas


3D Reconstruction

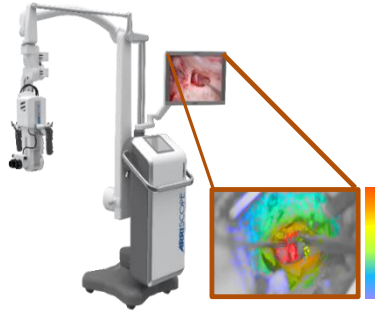




Scene Understanding / Multispectral Imaging



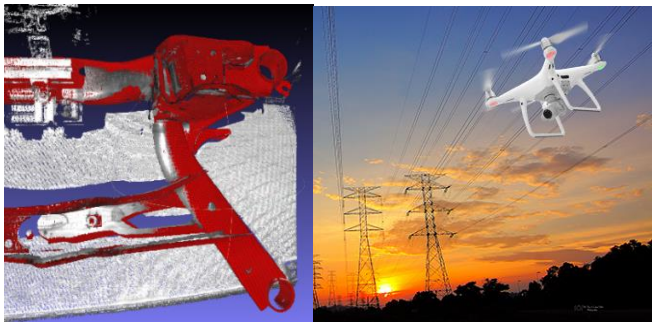
3D Tracking

Analysis and Synthesis of Humans



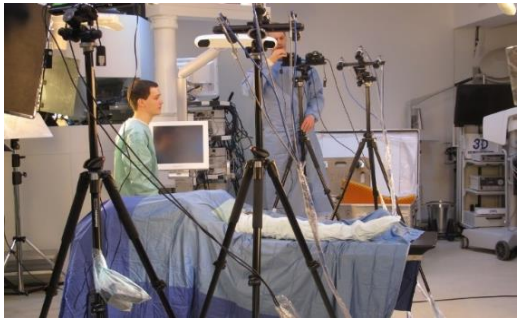
Augmented / Extended Reality

Application: Industry, Medicine, Security, Multimedia

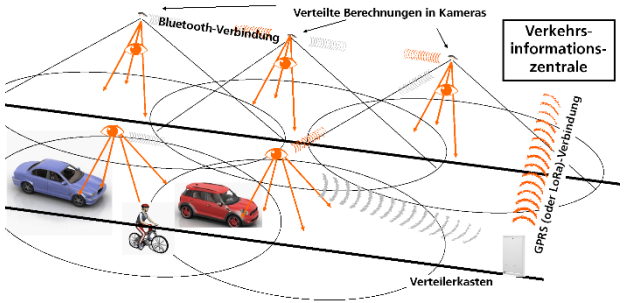


Industry

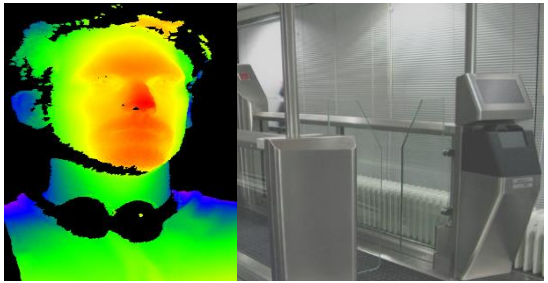
Construction



Medical Imaging



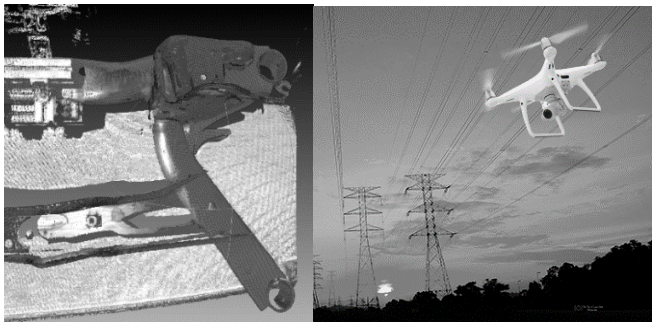
Mobility / Traffic Analysis



Security

Multimedia

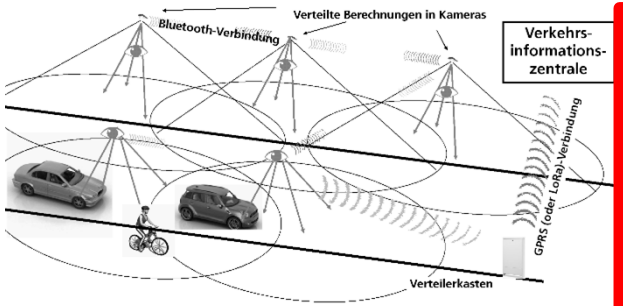
Detection of Attacks on Face Authentication/Verification



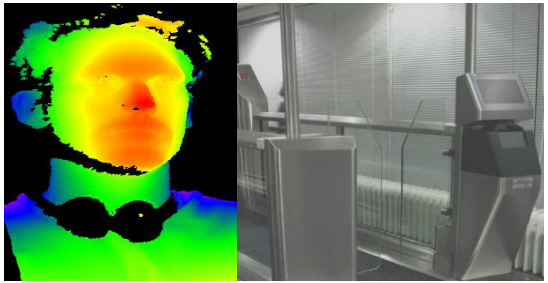
Industry

Construction

Medical Imaging



Mobility / Traffic Analysis



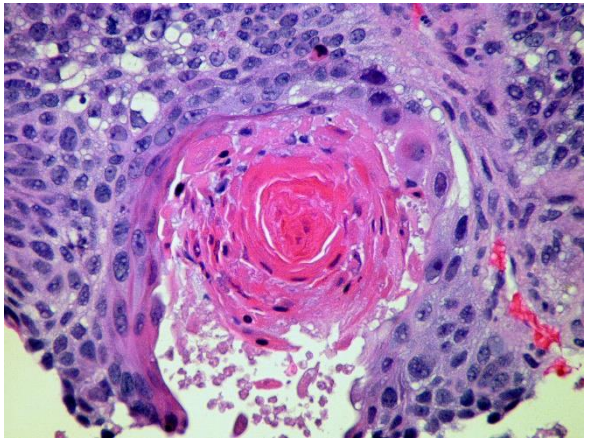
Security

Multimedia

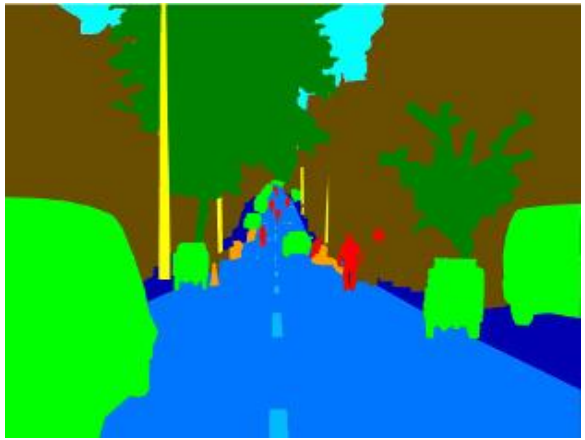
Schedule

- 100 min Presentation Trusted AI, Explainability, Attack Detection
- 10 min Break
- 40 min Hands on Session (Bias Detection, Explainability)

Success of Deep Learning



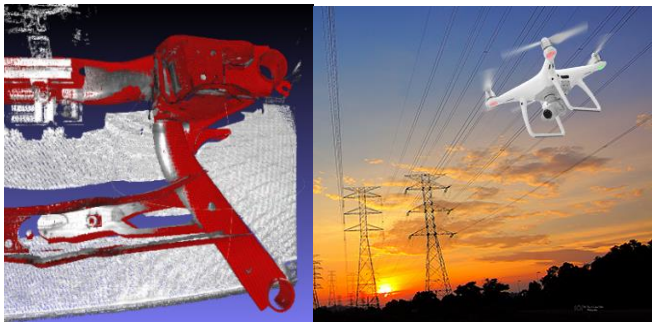
[Wikipedia]



[BBC]

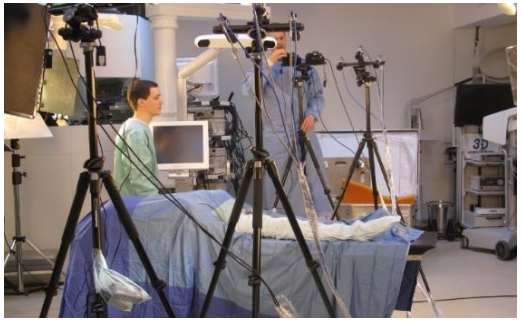
- High performance hardware (GPUs)
- Very large databases
- New network architectures

Correct Neural Network Decision Critical

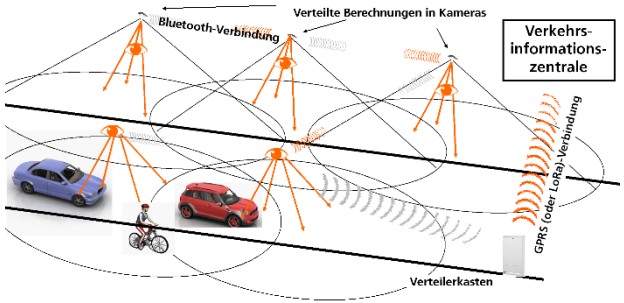


Industry

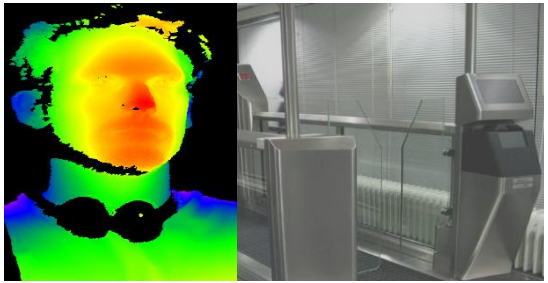
Construction



Medical Imaging



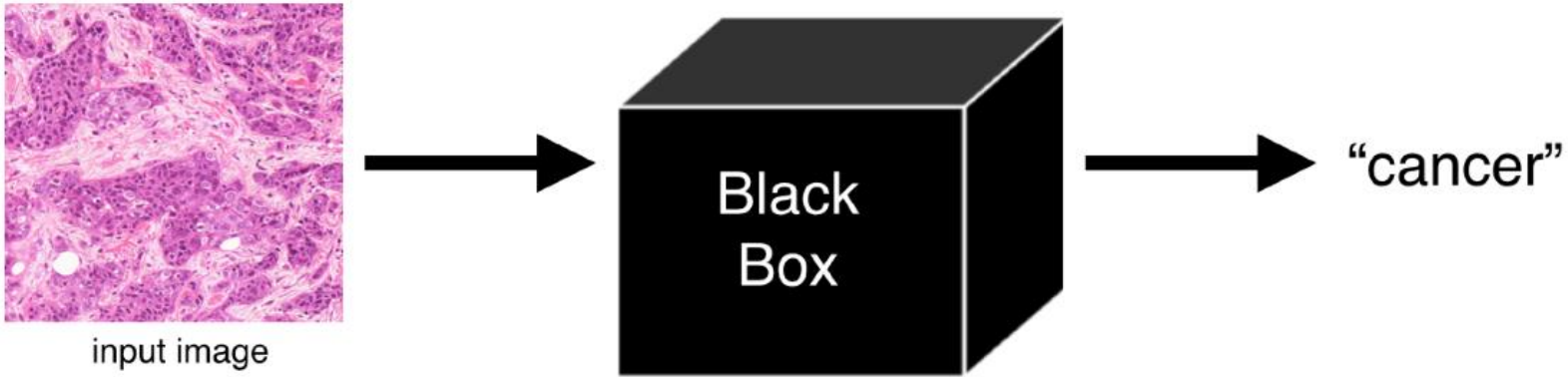
Mobility / Traffic Analysis



Security

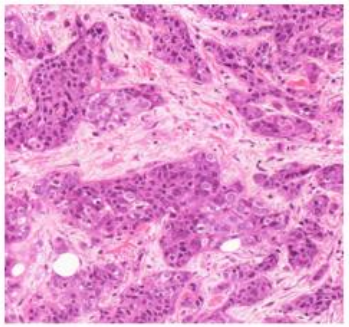
Multimedia

Are we sure that the network has learnt the right thing?



Clever Hans Principle

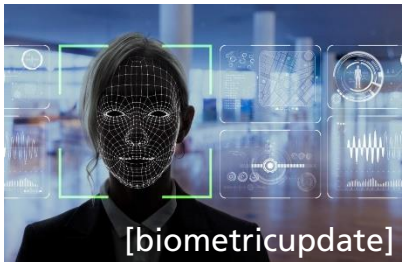
Explainability to Increase Trust in AI Systems



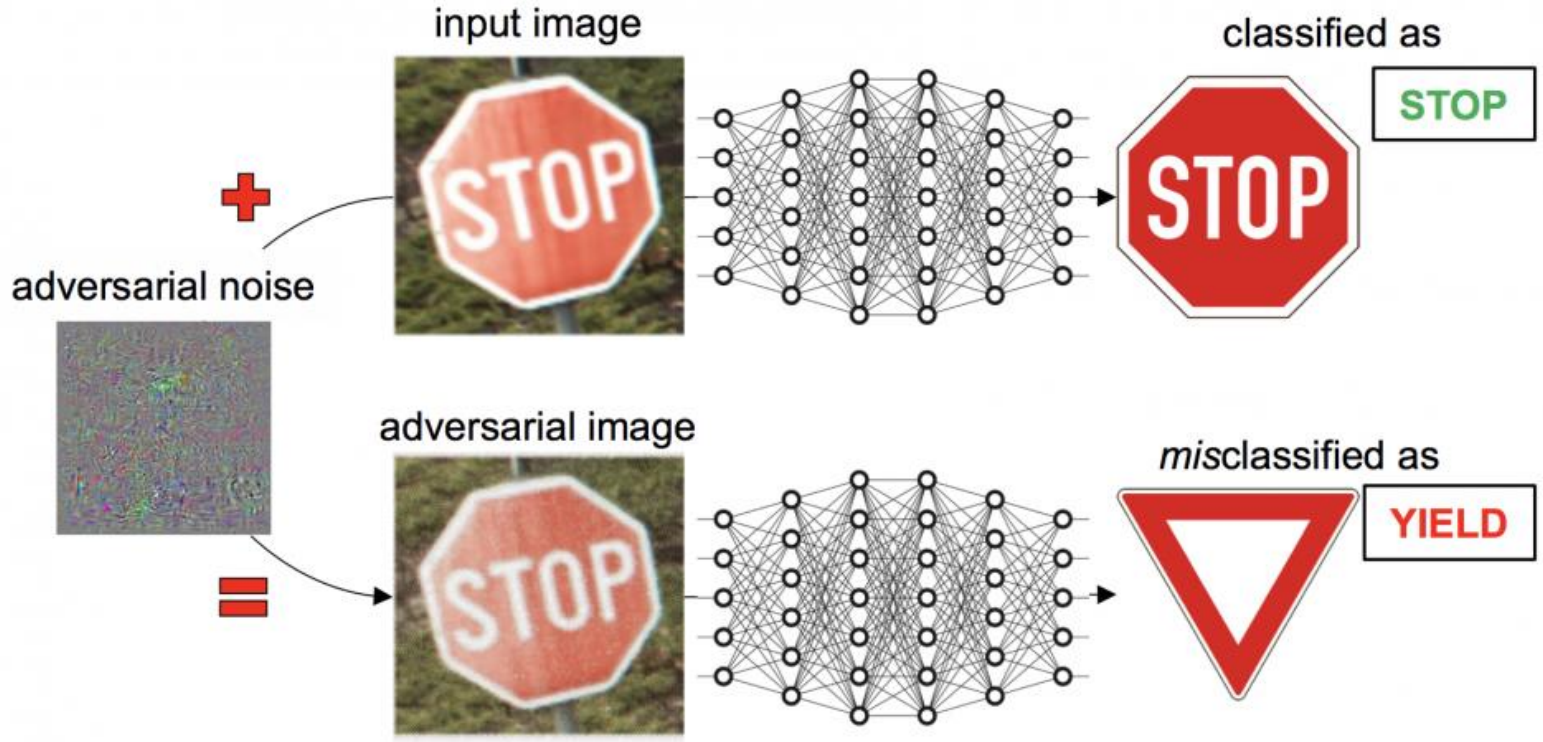
- Has the network learned the right thing?
  - Clever Hans principle?
  - Bias in data or network?
  - Robustness / ability to generalize on unseen data?

[Wikipedia]

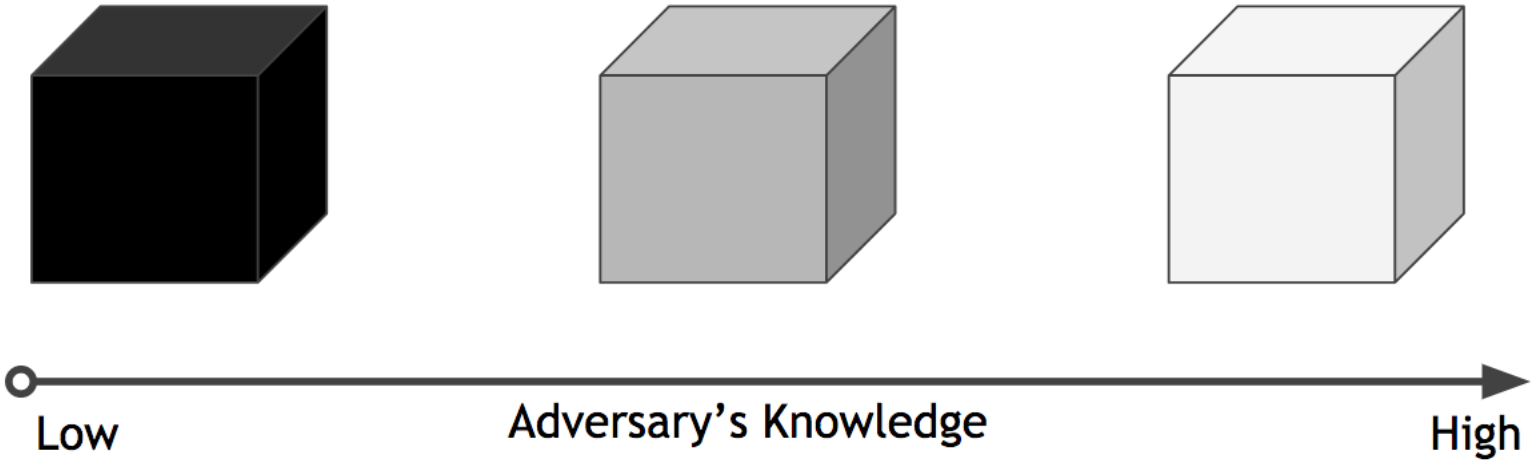
- Vulnerable to attacks on DNN ?
  - security
  - autonomous driving


[biometricupdate]

Fooling Neural Networks: Adversarial Attacks



Adversarial Attacks

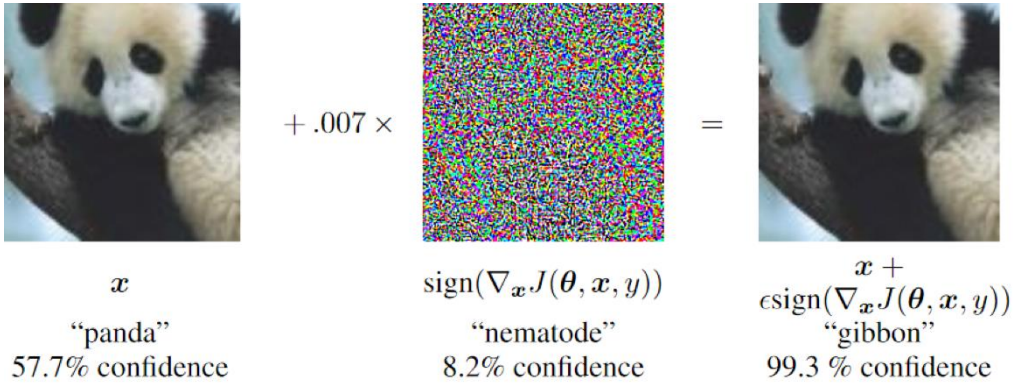


- black box
- Only network decisions known

- grey box
- Knowledge about network structure (but not weights)

- white box
- Full access to network

Adversarial Attacks



- Goal: $x' : D(x, x') < \vartheta, f(x') \neq y$
- Fast gradient sign method [Goodfellow2014]

$$\mathrm{x}' = x + \epsilon \cdot sign|\nabla_x J(\theta, x, y)|$$

- Targeted attack (particular fake result) by using appropriate gradient
- Extensions: iterative optimization, momentum based descent
- GANs for creating adversarial noise

- Very sparse high dimensional space, training data is concentrated in a small manifold

Carlini Wagner Attack

adversarial noise class image

- Optimization:

$$\text{minimize } ||\delta||_p \text{ subject to } C(x+\delta) = t, \quad x+\delta \in [0,1]^n$$

- New function f()
  <=0 iff C=t

$$\text{minimize } ||\delta||_p \text{ subject to } f(x+\delta) \leq 0, \quad x+\delta \in [0,1]^n$$

- Suggested f:
  (=0 if class t)

output before softmax

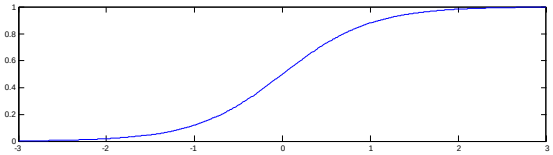
$$f(x) = \left([\max_{i \neq t} Z(x)_i] - Z(x)_t\right)^+$$

- Optimization:

$$\text{minimize } ||\delta||_p + cf(x+\delta) \text{ subject to } x+\delta \in [0,1]^n$$

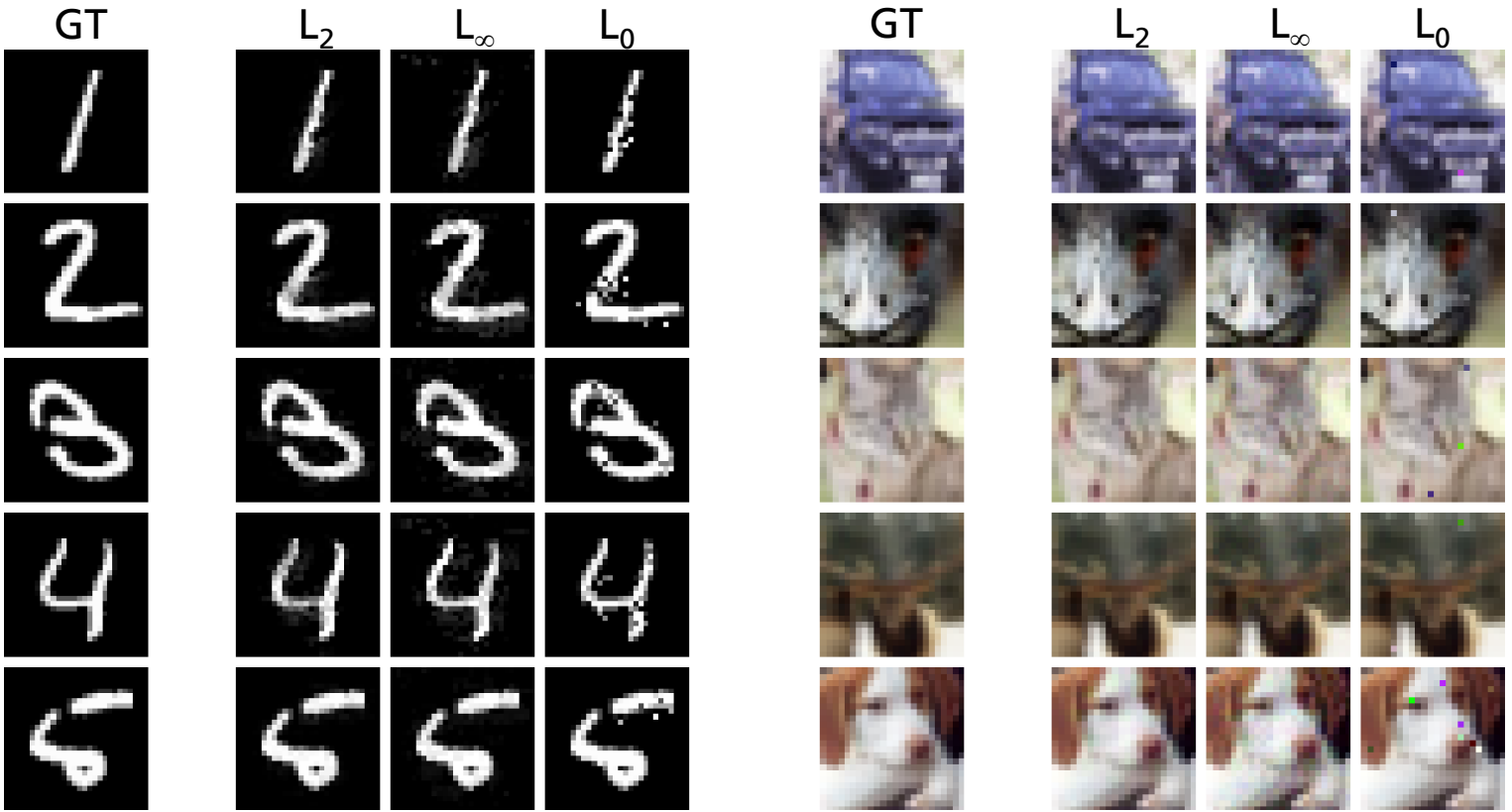
- Clipping to [0..1]:

$$\delta_i = \frac{1}{2}(\tanh(w_i)+1) - x_i$$



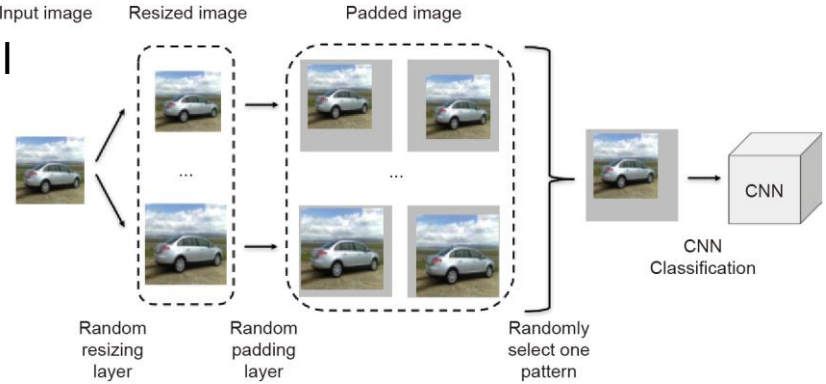
- Very powerful attack, works with different norms: L_2, L_0, L_∞

Carlini Wagner Attack [Carlini2017]



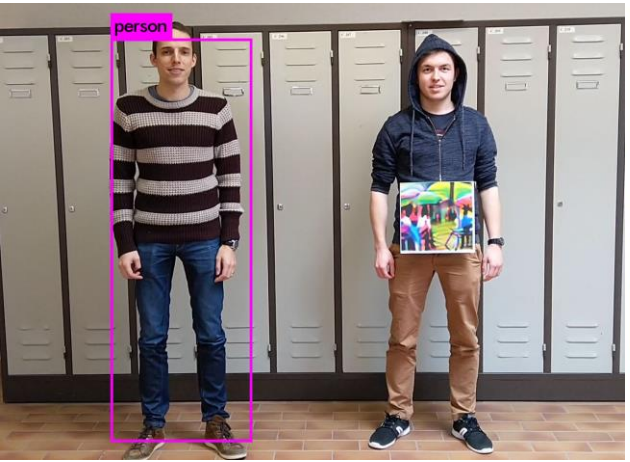
Defenses against Adversarial Attacks

- Adversarial learning
  - Generate adversarial attacks and train network with genuine and attack images
  - Can be done iteratively
  - Simple, computationally complex, works well



- Defensive Distillation
  - learn another network that outputs class probabilities with smoother gradients
  - softmax with “temperature” for more uniform p
- Adversarial noise detection
- Data preprocessing (resizing, blurring, denoising, JPEG, autoencoders, GANs, etc.)

Real World Examples – Adversarial Patches fool Networks



Person detection

[Thys2019]

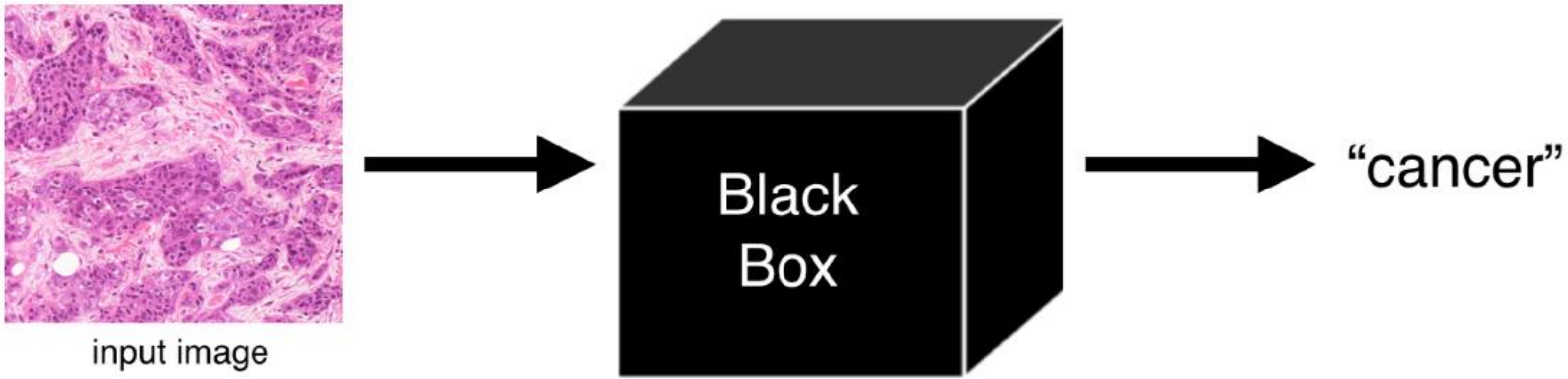
Face recognition

[Sharif2016]

Sign classification

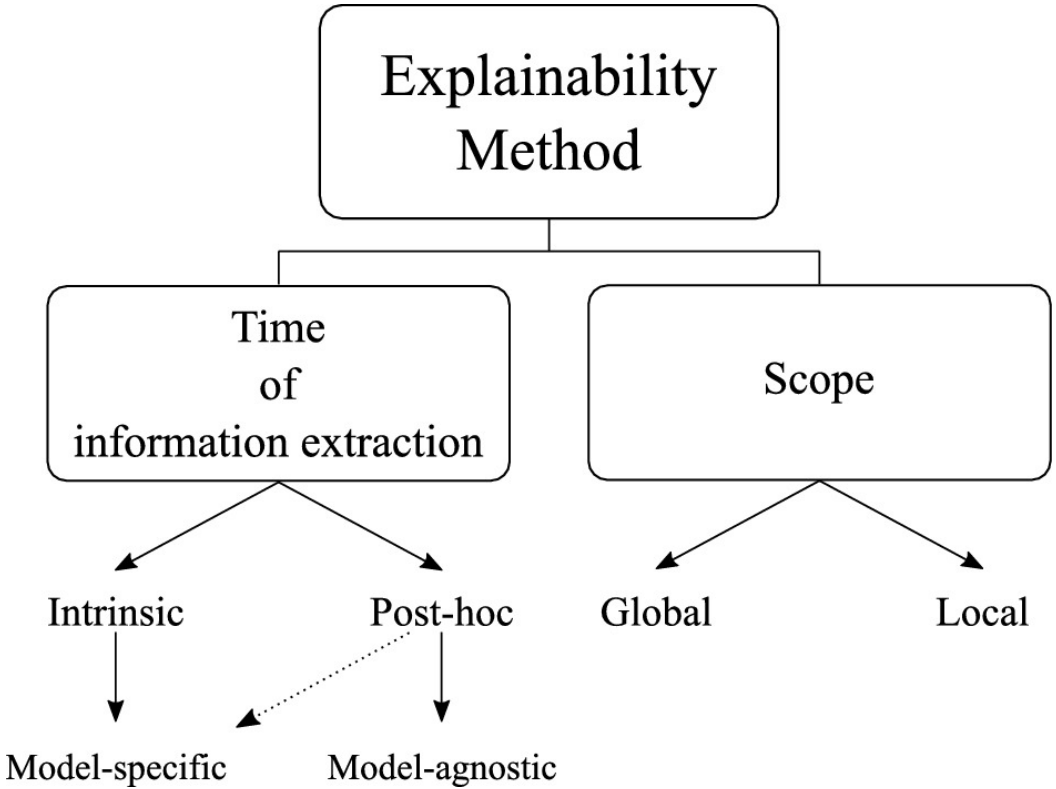
[Eykholt2018]

Explainability for Increased Trust in AI



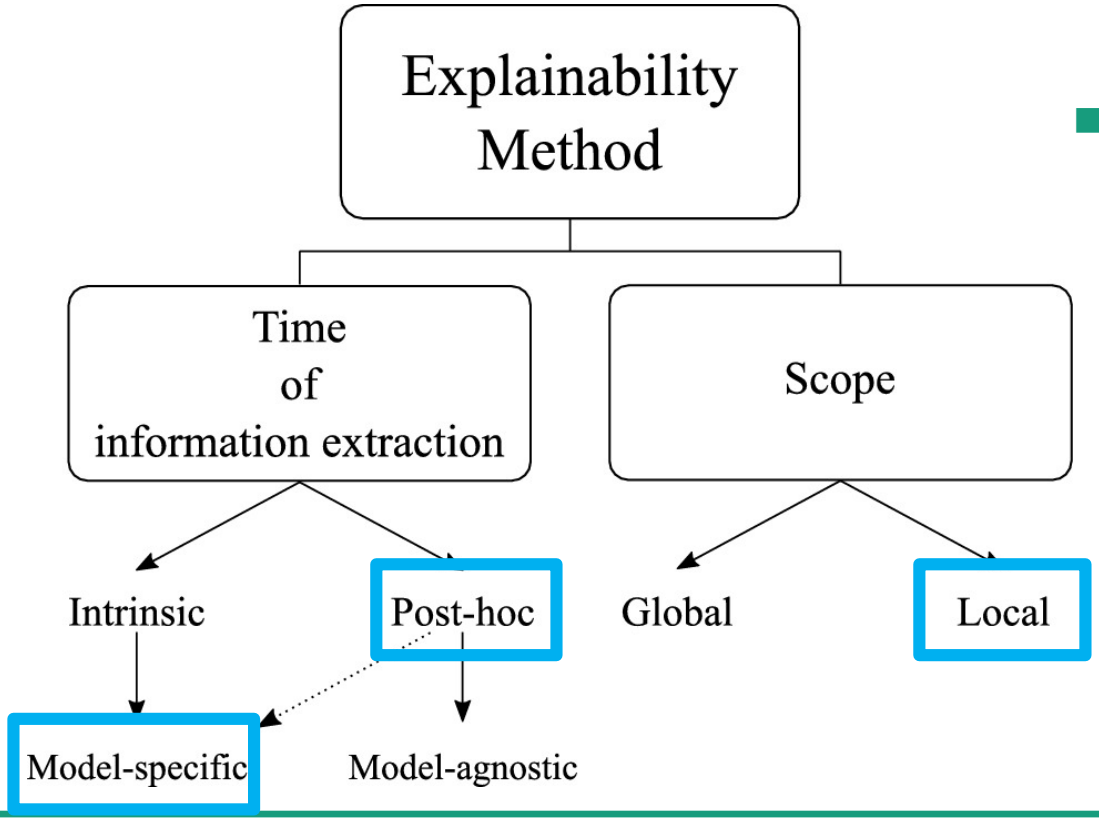
- Neural networks have shown great success in most computer vision tasks
- Very deep structures are difficult to interpret
- Black box decisions problematic for many security/safety related applications
- Interpretability and explainability can help increasing trust in AI decisions

Types of Explanation Methods



[Puiutta2020]

Types of Explanation Methods



- Focus on
  - CNNs
  - Image analysis
  - Security applications

[Puiutta2020]

Person Authentication / Verification

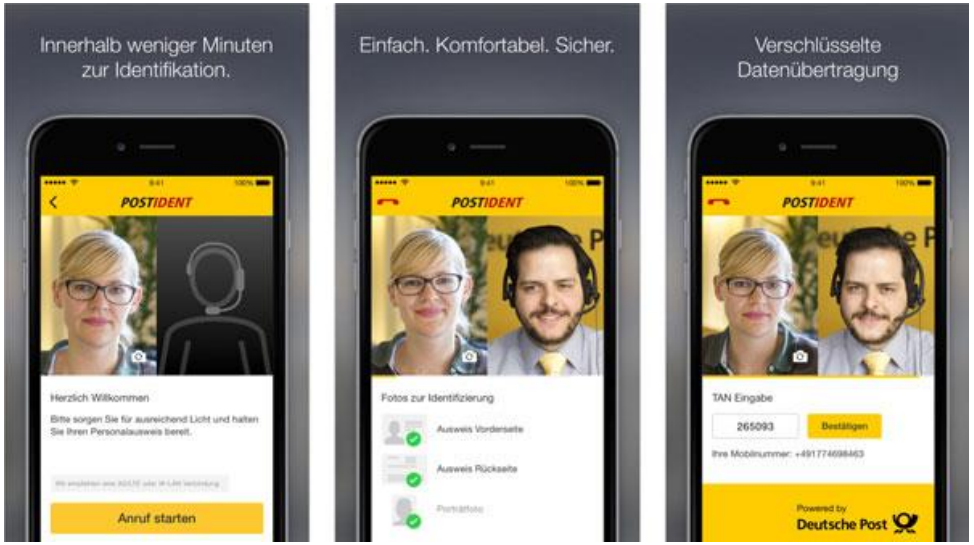
Automated border control

kiosk solutions

access control

Person Authentication / Verification

login for computers
smartphones



mobile / video identification

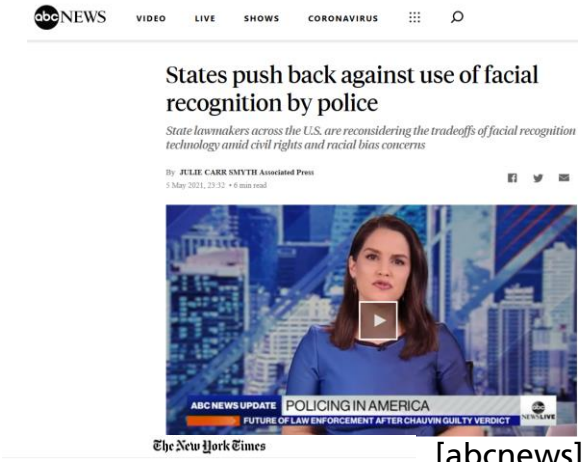
Face Recognition

[BDR]

- Face as a biometric feature
  - contactless, capturing from a distance
  - user friendly, widely accepted
- Challenges: pose, illumination, expressions, makeup/beard/glasses, age,…

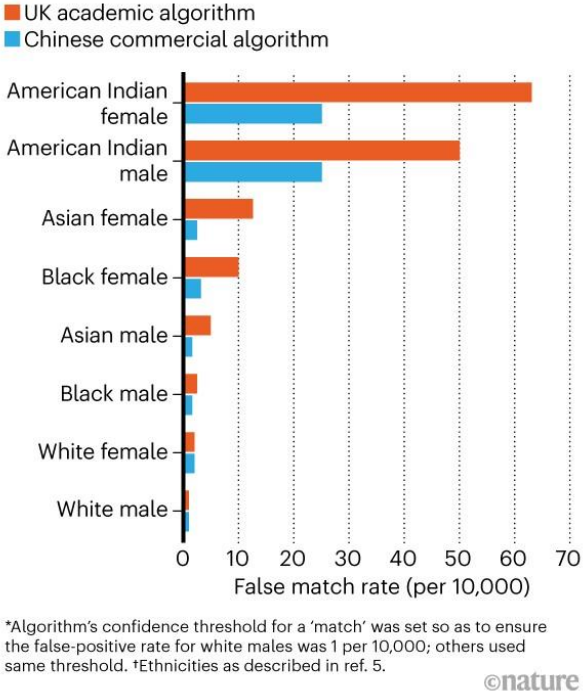
Bias in Face Recognition

[theregreview.org]



[abcnews]

[NewYork Times]



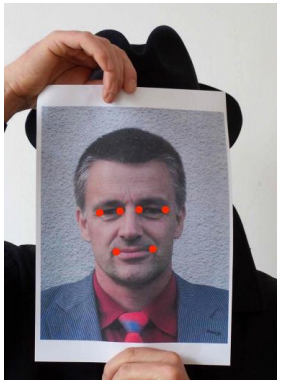
Attacks on Face Recognition Systems

- **Face morphing attacks**
- Provision of manipulated face reference image

- **Presentation attacks**
- Alteration of own appearance by masks, pictures etc.

- **Deep fake attacks**
- Manipulation of video for identification (video ident)

Presentation Attacks by Images, Videos, 3D Prints



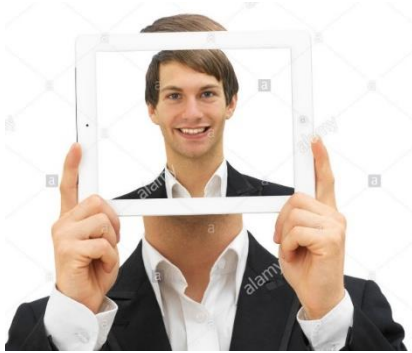
presentation

of pictures

[thatsmyface.com]

3D printouts

[broadsheet]



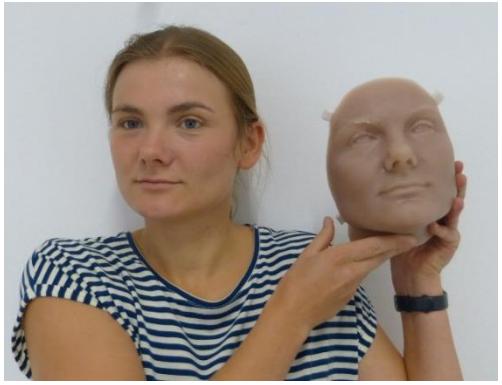
videos on tablets

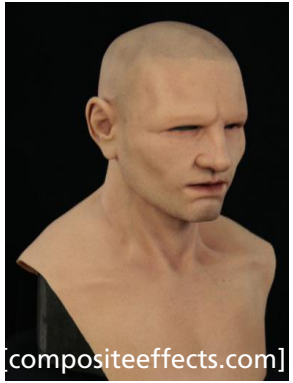
[alamy]

Presentation Attacks by Alteration of Own Appearance

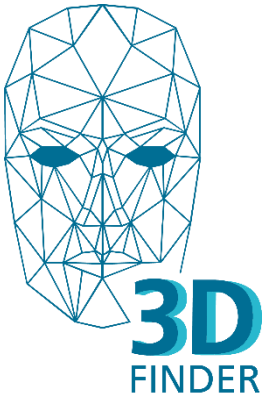
masks and partial masks

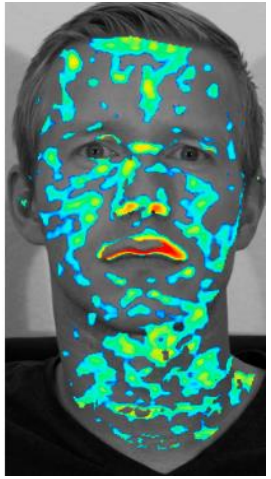
makeup / accessories



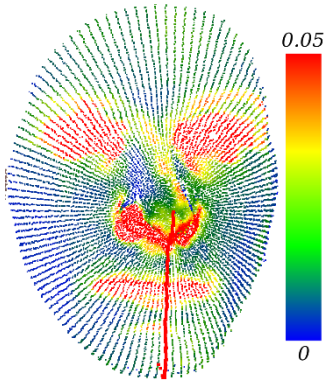


Presentation Attack Detection



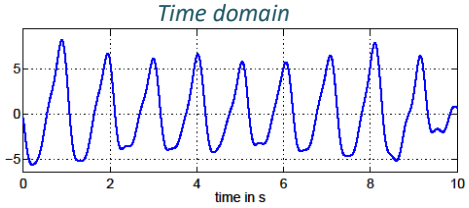


- Classification real face / fake by fusion of multiple detectors
  - 3D face geometry classification
  - Liveliness Detection using PhotoPlethysmoGraphy
  - Eye blink detection
  - Dynamic analysis of facial expressions

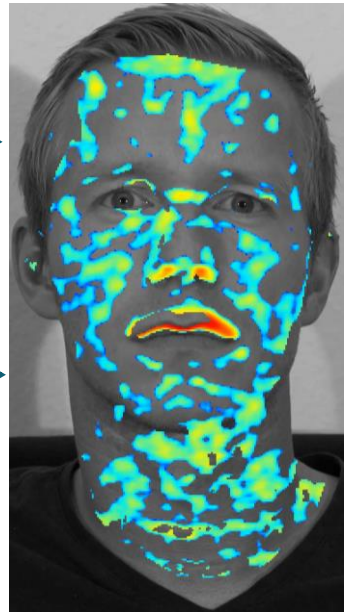


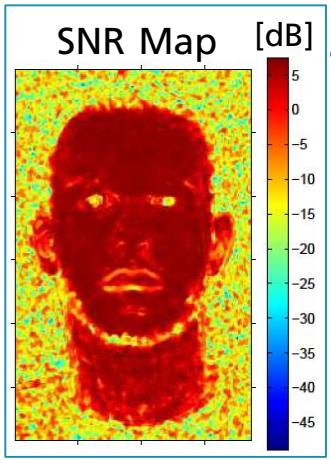
SPONSORED BY THE
Federal Ministry of Education and Research

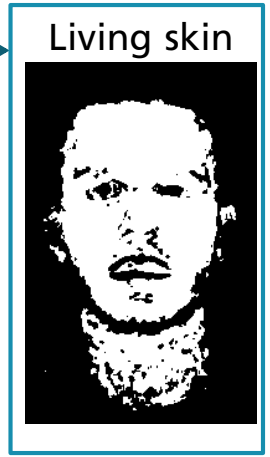
Liveliness Detection using PhotoPlethysmoGraphy



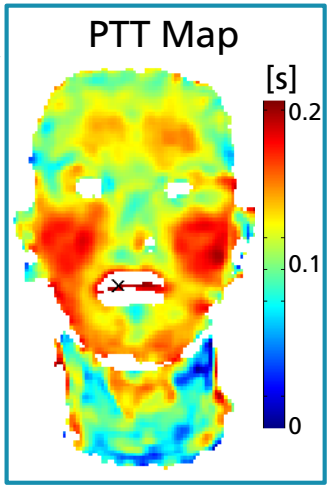
Pulse Rate = 58.6 BPM



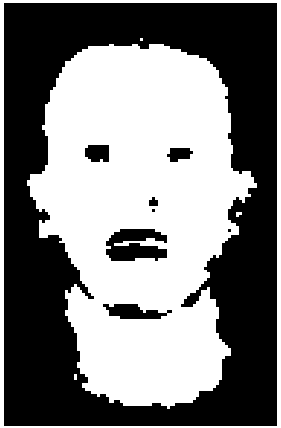


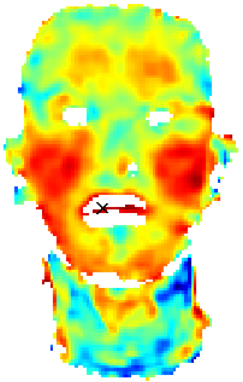


[Kossack, VMV2019]



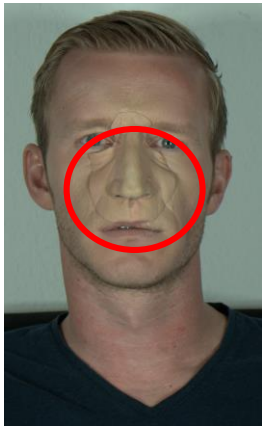
Detection of Partial Masks

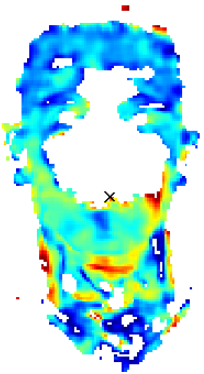




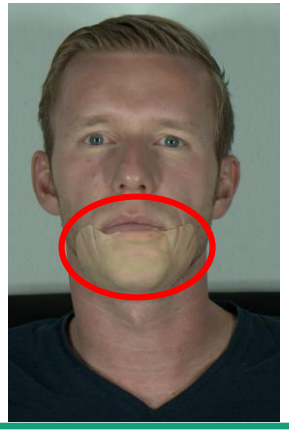
no mask

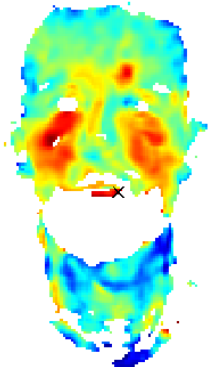
partial mask around nose





partial mask at chin





Classification of Face Images

female

glasses

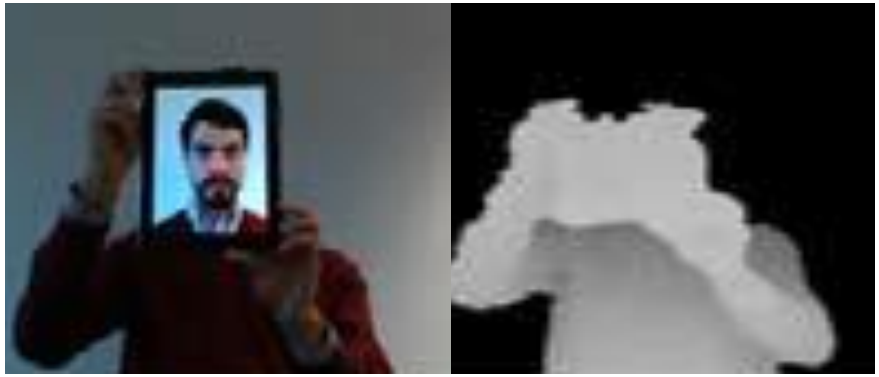
mustache

beard

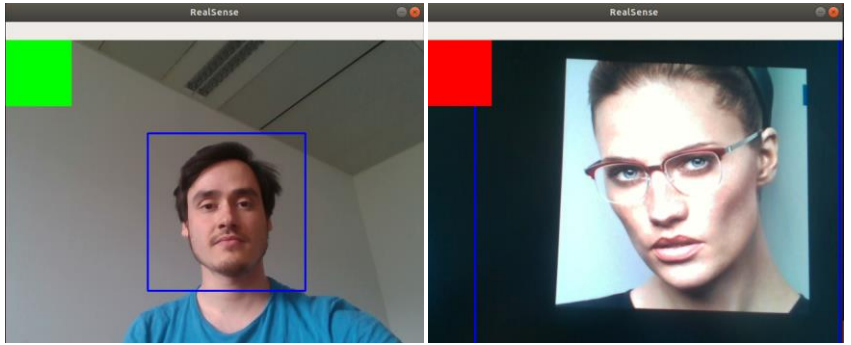
- Deep Learning for classification of face images (gender, glasses, mustache, beard)
- Trained on about 20000 labeled face images (several public databases + google)

class	accuracy
gender	98.4%
glasses	99.9%
mustache	98.1%
beard	97.8%

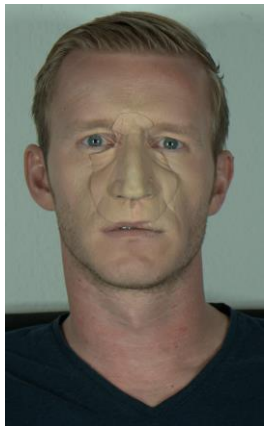
3D Face Geometry Classification



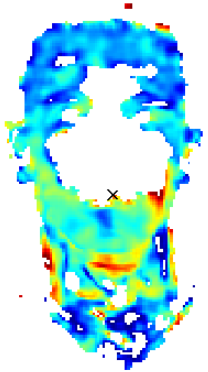
- Input: high quality 3D sensor & RealSense
- PAD: pictures, tablets
- Normalization of 3D face data
- Classification on depth maps
- Trained DNN (VGG19)
- 4 false positives from 631 samples



How Robust are the Classifiers?



masks



face attributes

face geometry

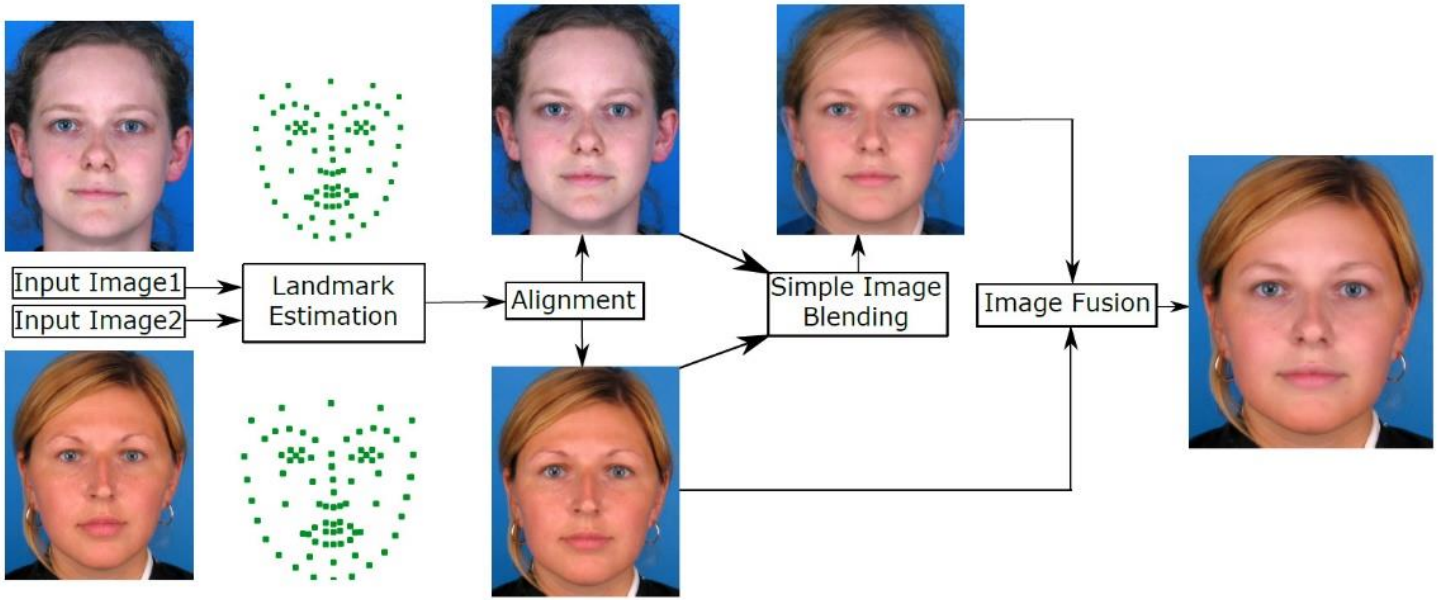
Face Morphing Attacks


Originals


Morphs

- Creation of “average” face image from two or more individuals
- Face morph contains characteristics of both individuals
- Face recognition tends to accept both individuals when comparing against morph

(Automatic) Creation of Face Morphs



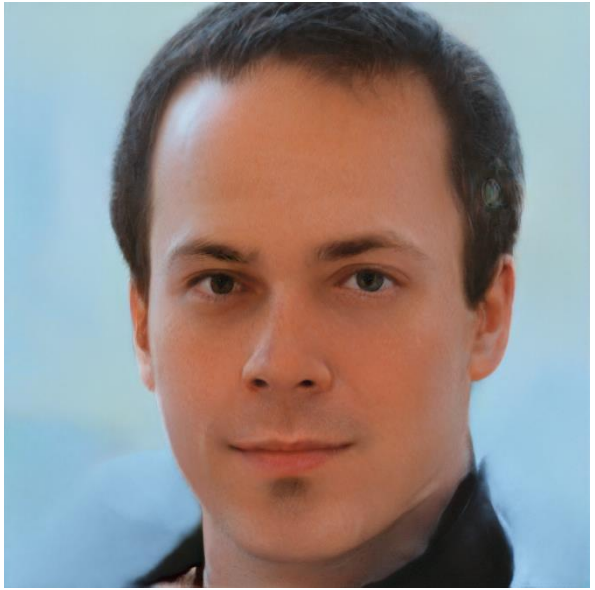
- Geometric face alignment
- Image blending
- Fusion of face morph into one original image to avoid artifacts in hair / background

GAN based Face Morph Generation



Modification of Face Properties using GANs

original



younger

older

Face Morphing Detection

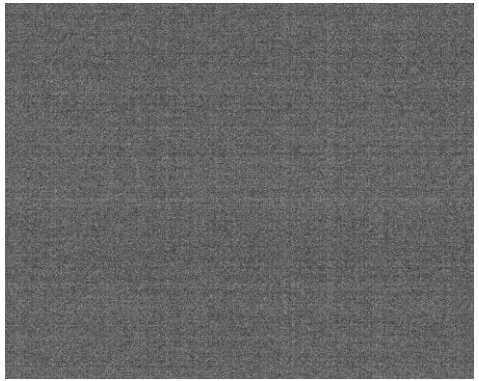
- **D4Fly**: Detecting Document frauD and iDentity on the fly

- Duration 2019-2022

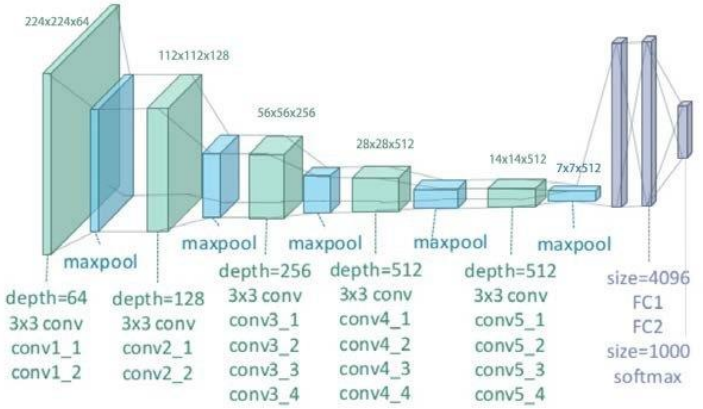
VERIDOS
TRILATERAL RESEARCH
TNO
Regula
Home Office
Fraunhofer Heinrich-Hertz-Institut
DEMOKRITOS
BPTI BALTIC INSTITUTE OF ADVANCED TECHNOLOGY
raytrix
OVD KINEGRAM A KURZ Company
Immigration and Naturalisation Service Ministry of Justice and Security
WAT
NTNU
University of Reading
VTT
OAΠ PIRAEUS PORT AUTHORITY S.A.
Royal Netherlands Marechaussee
THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION HORIZON 2020

(Blind) Face Morphing Attack Detection

- Low level information
  - Camera sensor noise
  - Double (JPEG) compression
- Signal information
  - Blurring
  - Gradient statistics
- High level
  - Alignment / ghosting artifacts
  - Implausible face content
    - Double eyelashes
    - Asymmetries
    - Inconsistent highlights



Blind Detection of Face Morphing Attacks

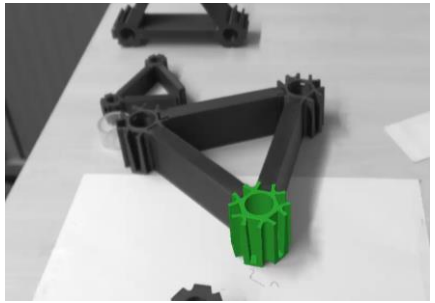


- Morphing attack detection using Deep Neural Networks
- Creation of training and test datasets, >2000 original images
- Image pre-processing (filtering, noise) to increase variations in dataset
- Comparison of different network architectures, VGG19, GoogLe,…
- Equal error rate (EER): 3%

[Seibold, IWDW 2017]

Deep Fake Attacks on Video Ident Systems

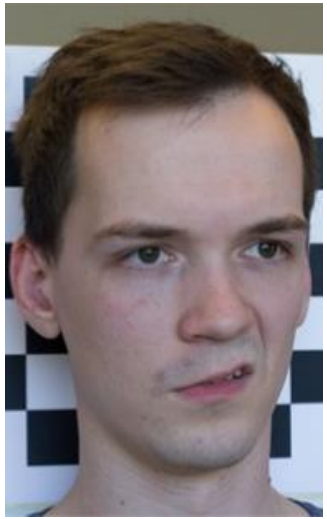
- Verification of identity via video chat and presentation of passport
- Used e.g. for applications for a bank account
- Attacker can manipulate video
  - Modify the passport
  - Replace face region by other identity (retargeting)



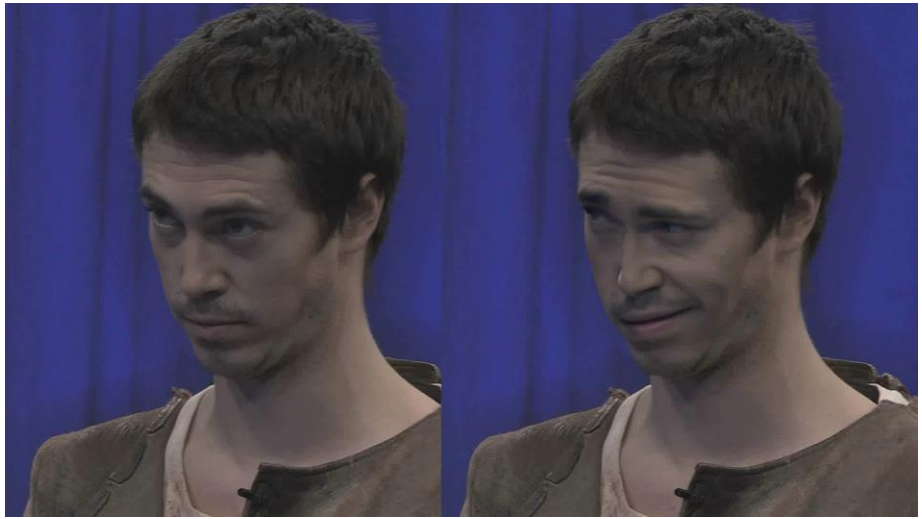
Fake-ID

- BMBF funded project for the detection of deep fakes for video ident
- Duration 2021-2023
- Detection of video manipulation
- Explainability for evidence in court

Face Retargeting



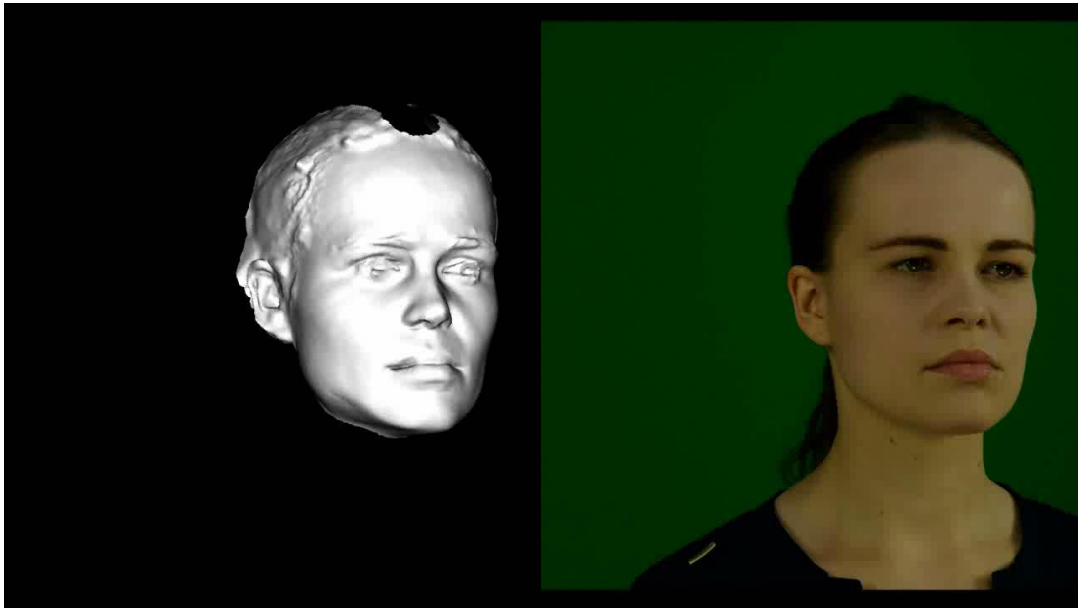
source identity



target identity target + source expressions

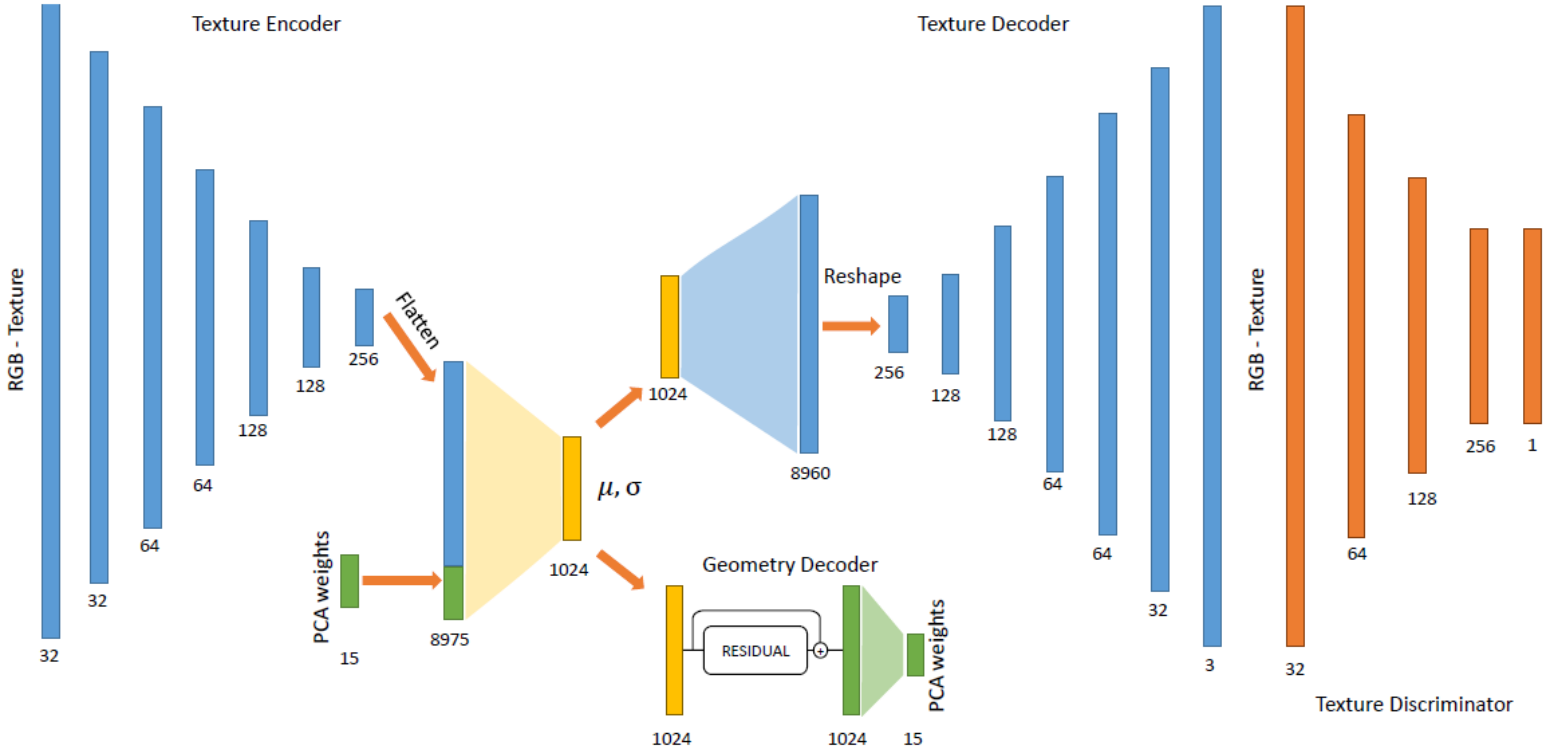
- Map head and face motion / expressions of source to target person
- Animation can be driven from video, feature points, speech, text etc.
- Synthesis of entire video or only facial region (using outline from original video)

Face and Expression Tracking



- Rough face pose and facial expression tracking (also be used for fake analysis)
- Details and realism by addition texture layer (autoencoder / GAN)

Variational Autoencoder for Synthesis of Texture & Geometry



[Paier, CVMP2020]

Synthesis of Facial Expressions

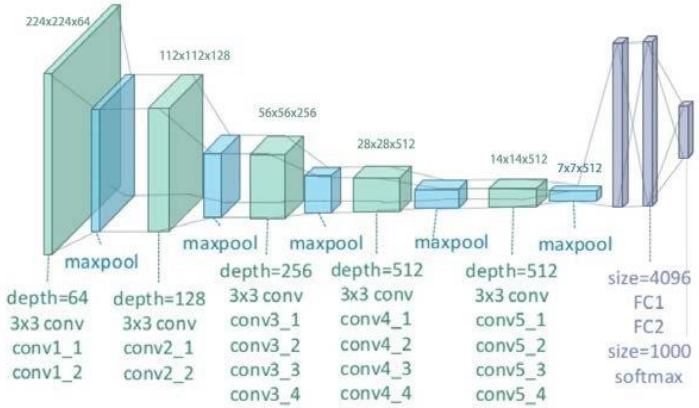
synthesized from face video

original video

synthesized from text
(without head motion)

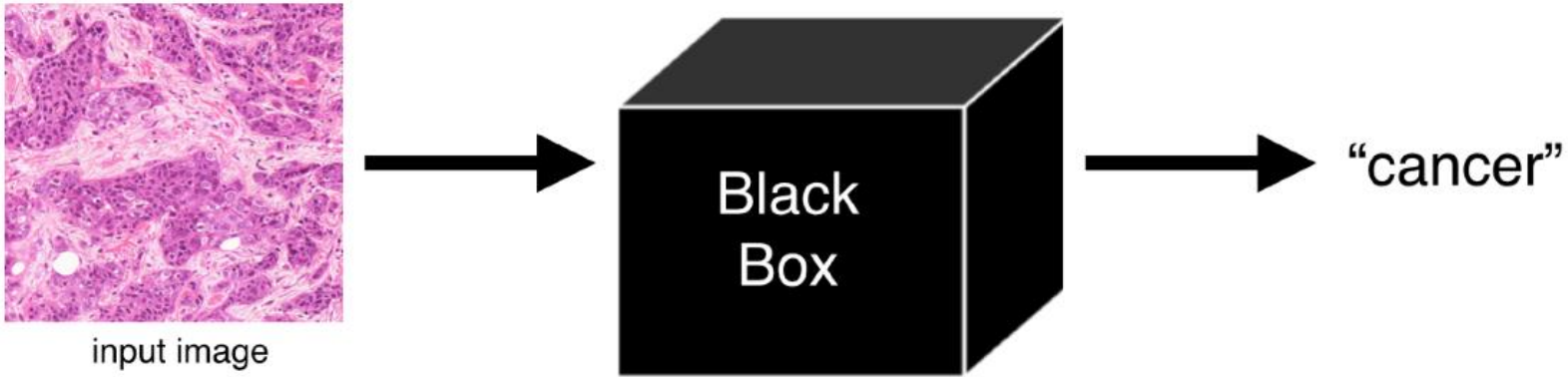
Deep Fake Detection (under development)

- Track 3D pose and expressions of video
- Align frames
- Train DNN on spatially aligned video sequence
- Analyze temporal motion data



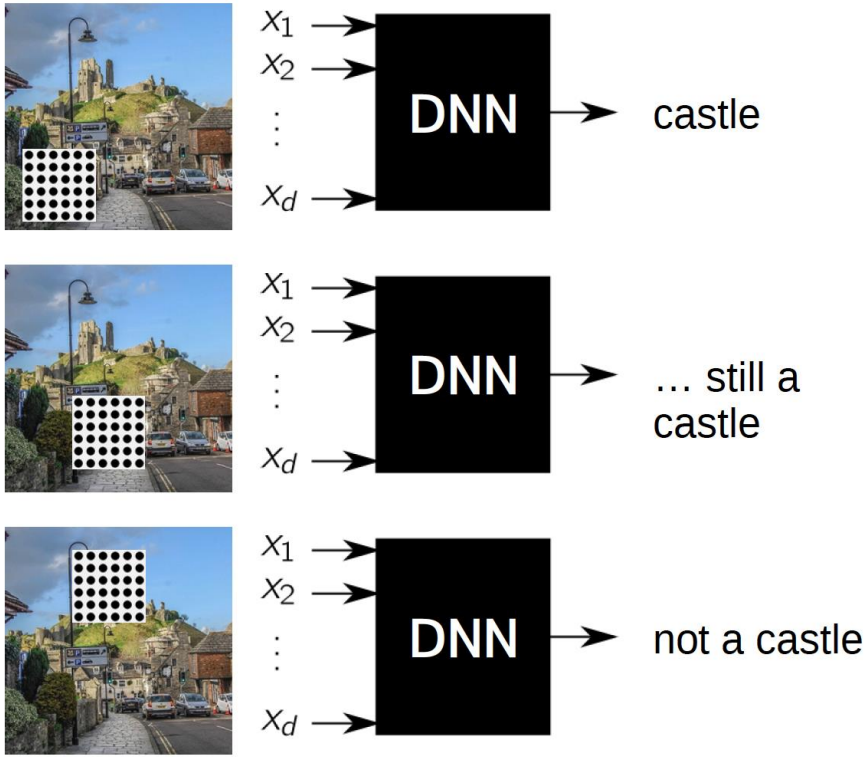
- What has the classifier actually learned?
- How can we prove to someone that a video is a fake?

Are we sure that the network has learnt the right thing?



Interpretability: Occlusion / Perturbation

- Assess feature relevance by testing the model response to their removal or perturbation
- Disadvantages
  - slow
  - assumes locality
  - perturbation may introduce artifacts
- Optional inpainting:



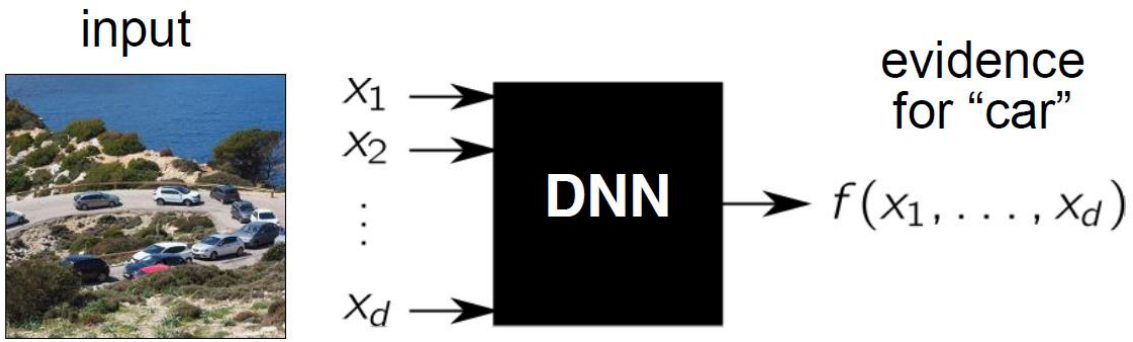
Local Interpretable Model-Agnostic Explanations (LIME)

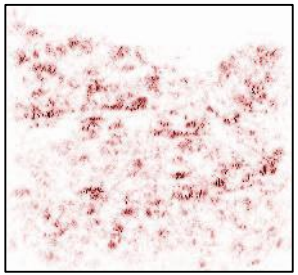


[Ribeiro2016]

- Perturb inference input and compute output of black box model
  - For images: segment into super pixels and flip them
- Train an interpretable classifier (linear, decision tree) that describes the behavior of the black box model in a local neighborhood
- Present most important superpixels as explanation or interpret local model
- Human friendly explanation, model agnostic but instable results, sensitive to “neighborhood”

Interpretability: Sensitivity Analysis



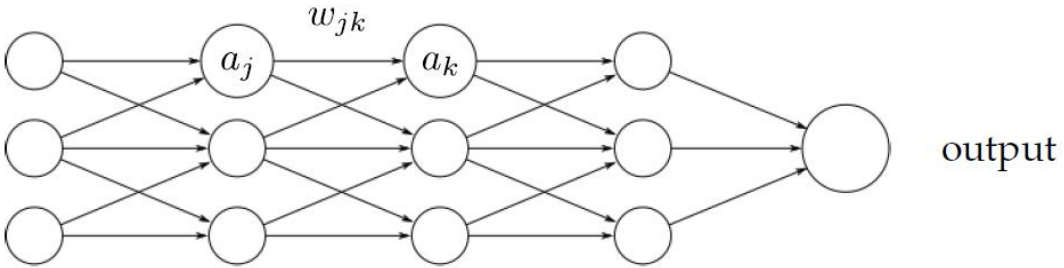


does not highlight cars

- Relevance of input feature i: $R_i = \left(\frac{\partial f}{\partial x_i}\right)^2$
- Sensitivity analysis explains a *variation* of the function, not the function value itself
- Input gradients become unreliable with increased network depth

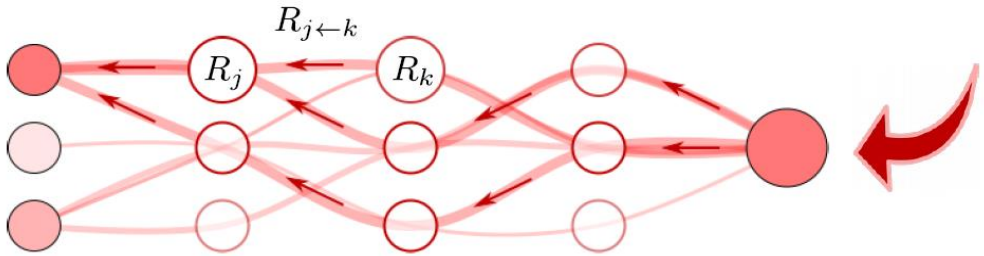
Interpretability: Layer-wise Relevance Propagation (LRP)

- Forward pass



- Relevance propagation

explanation



$$R_j = \sum_k R_{j\leftarrow k}$$

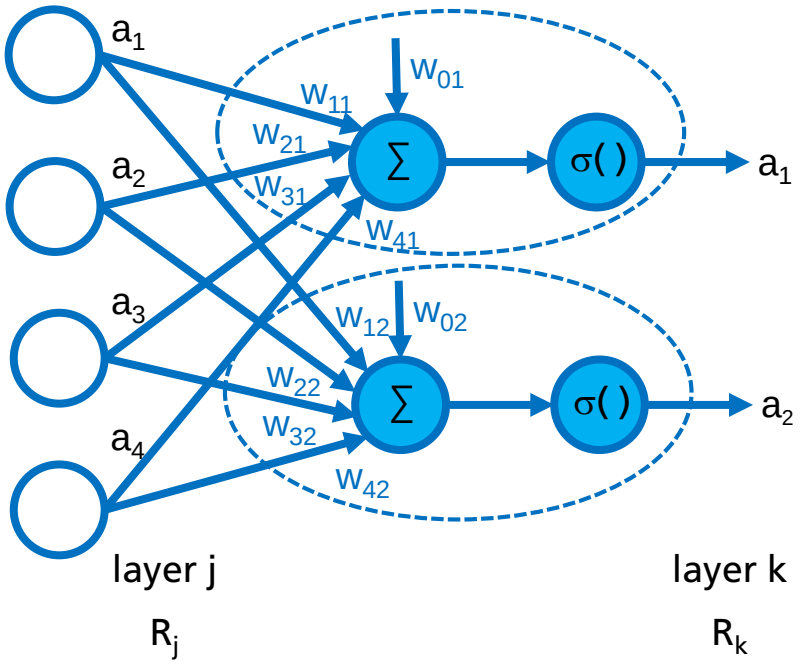
$$\sum_j R_{j\leftarrow k} = R_k$$

[Bach, On Pixel-wise Explanation …by LRP, PLOT ONE 2015]

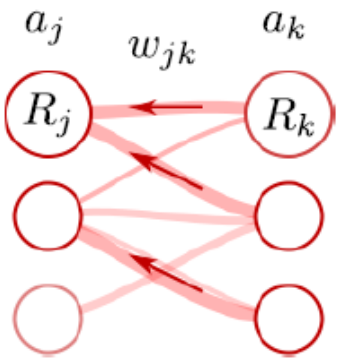
Relevance Propagation

- Conservation of relevance: $\sum_j R_j = \sum_k R_k$
- ReLU layer: $a_k = \max\left(0, \sum_{0,j} a_j w_{jk}\right)$
- Relevance propagation

$$R_j = \sum_k \frac{a_j w_{jk}}{\sum_j a_j w_{jk}} R_k$$



LRP Propagation Rules

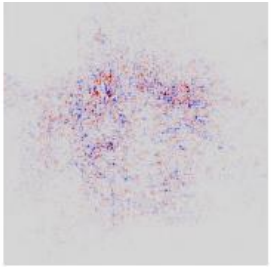


$$a_k = \max\left(0, \textstyle\sum_{0,j} a_j w_{jk}\right)$$

LRP-0

$$R_j = \sum_k \frac{a_j w_{jk}}{\sum_j a_j w_{jk}} R_k$$

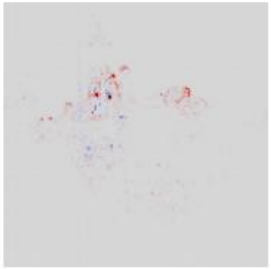
Equivalent to gradient x input, noisy



LRP-ϵ

$$R_j = \sum_k \frac{a_j w_{jk}}{\epsilon + \sum_j a_j w_{jk}} R_k$$

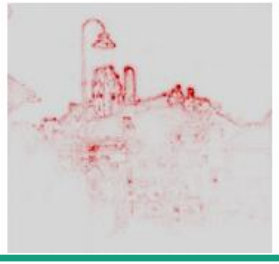
Reduces noise, increases sparsity



LRP-γ

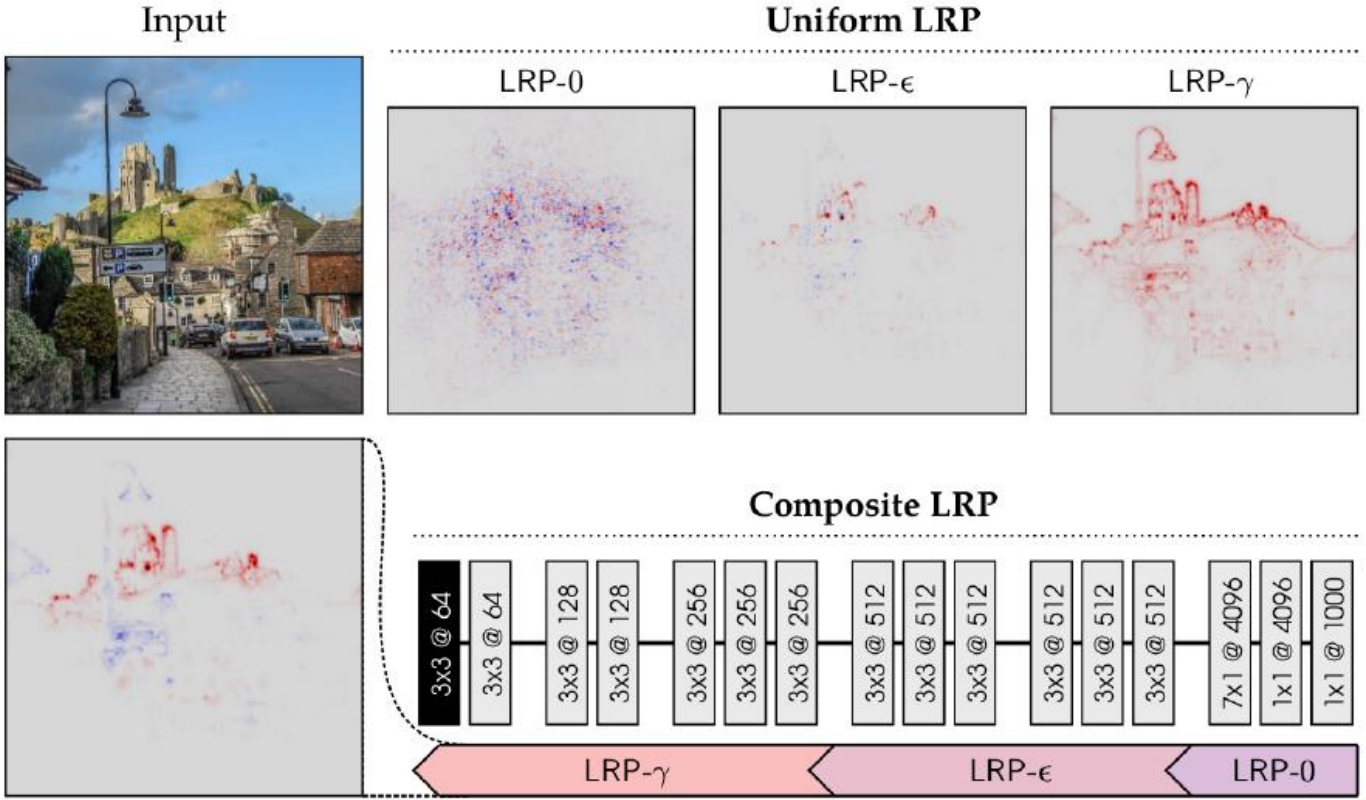
$$R_j = \sum_k \frac{a_j (w_{jk} + \gamma w_{jk}^+)}{\sum_j a_j (w_{jk} + \gamma w_{jk}^+)} R_k$$

Reduces noise, reduces sparsity



$w^+=\max(0,w)$

Different Rules at Different Layers



[Samek]

Special Rules

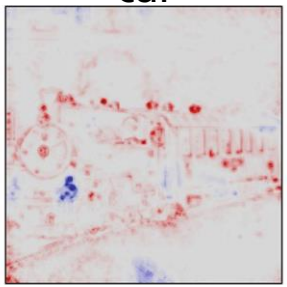
- Input layer: no ReLU, pos. + neg. activation

$$R_i = \sum_j \frac{x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_i x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-} R_j$$

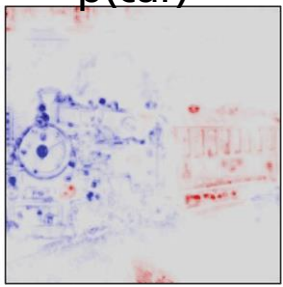
[l,….h]: range of pixel values

- Special layers
  - average pooling: only linear operation, can be incorporated into weights
  - max pooling: only “one flow” of relevance
  - batch normalization: only shift and scaling during testing
- Output layer, softmax:
  - $\mathrm{R} = z_c \cdot \exp(-\mathrm{z_c}) / \sum_{c'} \exp(-z_{c'})$
  - explain probability of a class instead of class its

car



p(car)



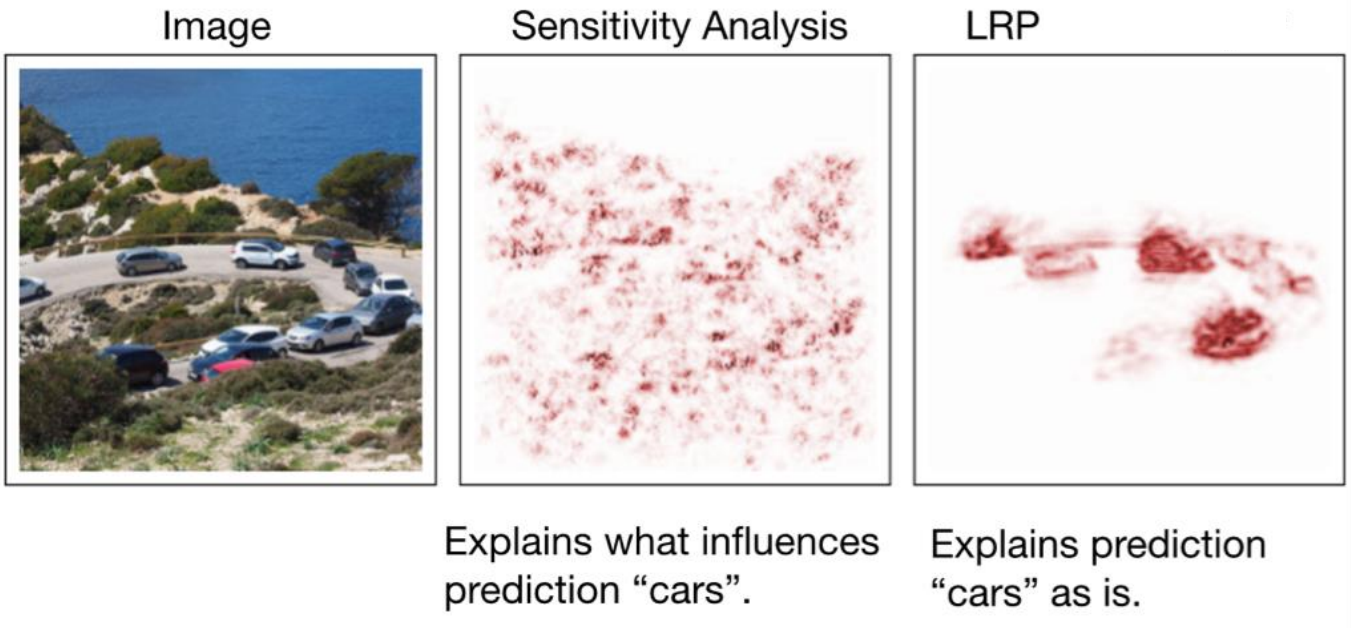
Rule Overview

Name	Formula	Usage	DTD
LRP-0 [7]	$R_j = \sum_k \frac{a_j w_{jk}}{\sum_{0,j} a_j w_{jk}} R_k$	Upper layers	✓
LRP-ϵ [7]	$R_j = \sum_k \frac{a_j w_{jk}}{\epsilon + \sum_{0,j} a_j w_{jk}} R_k$	Middle layers	✓
LRP-γ	$R_j = \sum_k \frac{a_j (w_{jk} + \gamma w_{jk}^+)}{\sum_{0,j} a_j (w_{jk} + \gamma w_{jk}^+)} R_k$	Lower layers	✓
LRP-$\alpha\beta$ [7]	$R_j = \sum_k \left(\alpha \frac{(a_j w_{jk})^+}{\sum_{0,j} (a_j w_{jk})^+} - \beta \frac{(a_j w_{jk})^-}{\sum_{0,j} (a_j w_{jk})^-} \right) R_k$	Lower layers	$\times^a$
flat [30]	$R_j = \sum_k \frac{1}{\sum_j 1} R_k$	Lower layers	$\times$
w^2-rule [36]	$R_i = \sum_j \frac{w_{ij}^2}{\sum_i w_{ij}^2} R_j$	First layer ($\mathbb{R}^d$)	✓
$z^{\mathcal{B}}$-rule [36]	$R_i = \sum_j \frac{x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_i x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-} R_j$	First layer (pixels)	✓

(^{a}DTD interpretation only for the case $\alpha = 1, \beta = 0$.)

[Samek2019]

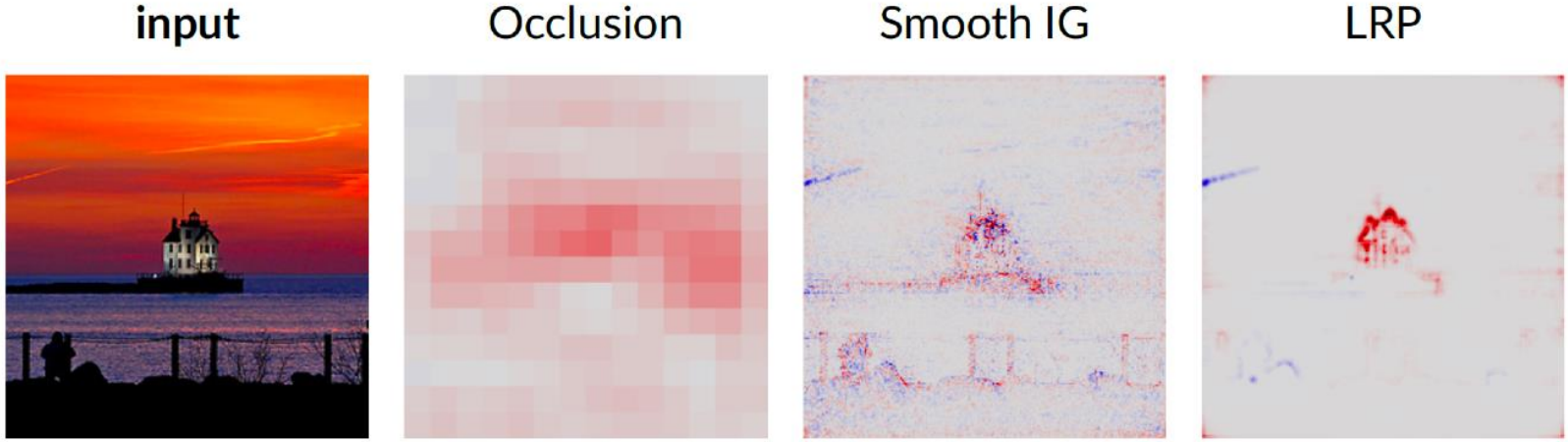
Explanation by Layer-wise Relevance Propagation



Try yourself: www.heatmapping.org

[Samek]

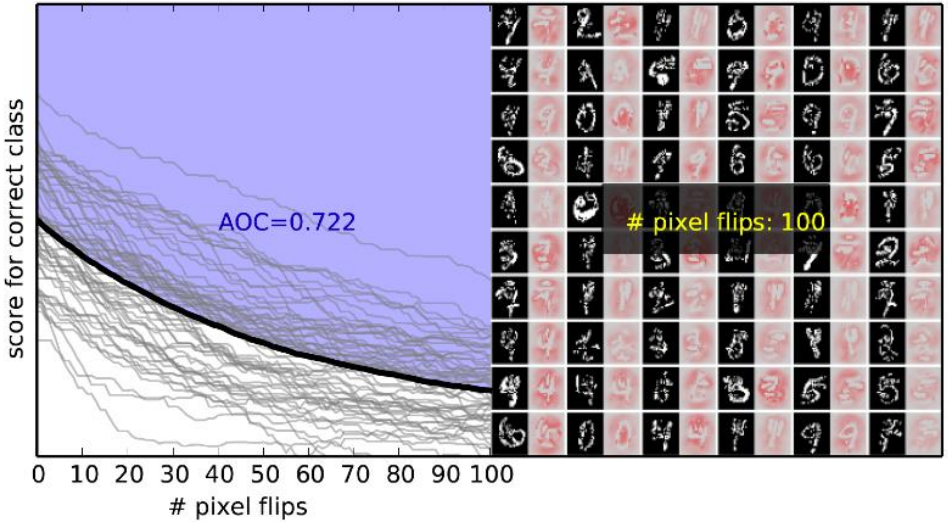
Other Backpropagation Methods for Explainability

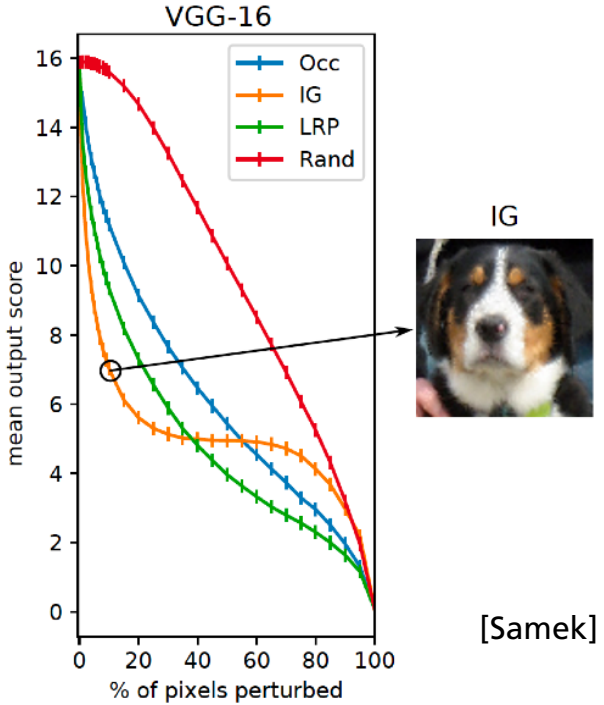


[Samek]

- Guided Backprop [Springenberg2015]: set all negative gradients to 0
- Integrated Gradients [Sundararajan 2017]: integrate gradients from baseline image (black) to current input image
- SmoothGrad [Smilkov2017]: average gradients in a small perturbation
- GradCAM [Selvaraju2019]: see later

Evaluating Explainability Methods

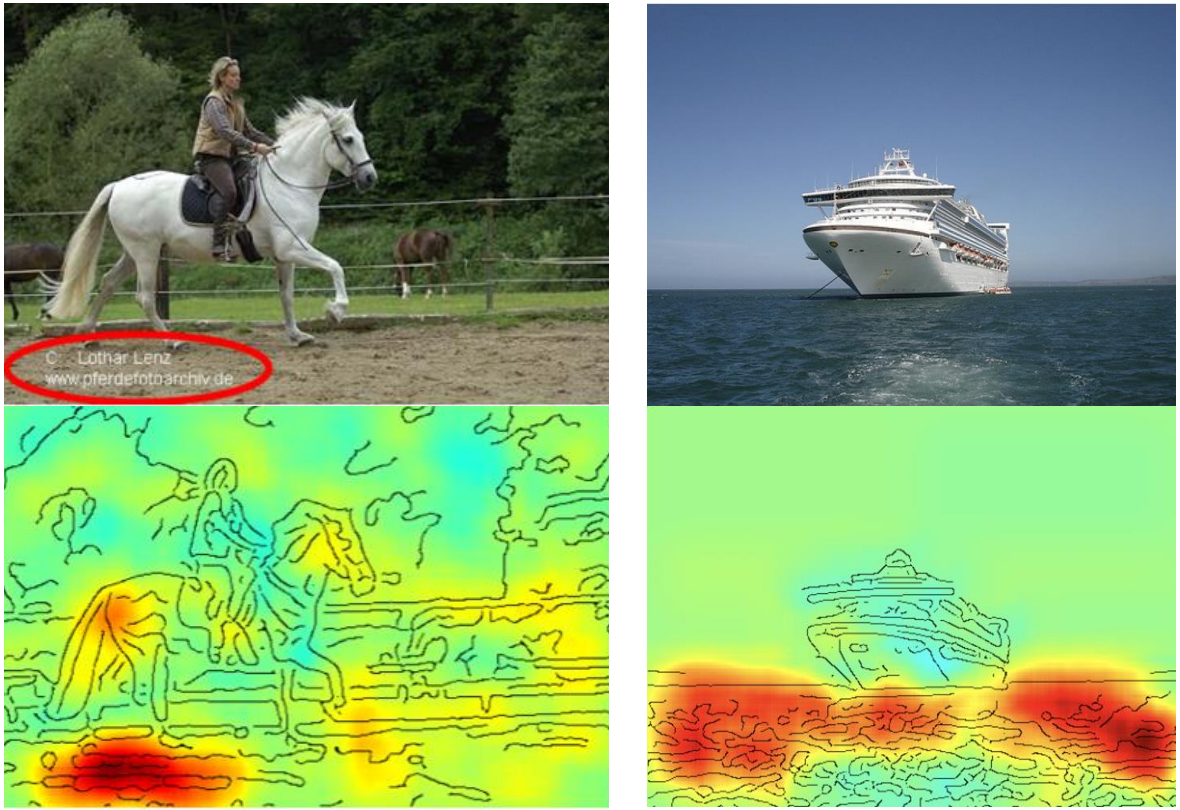




[Samek]

- Pixel flipping: destroy pixels in order of their importance with respect to explanation (filled with inpainting to avoid artifacts)
- Measure network accuracy
- The faster the output decreases, the better the explanation

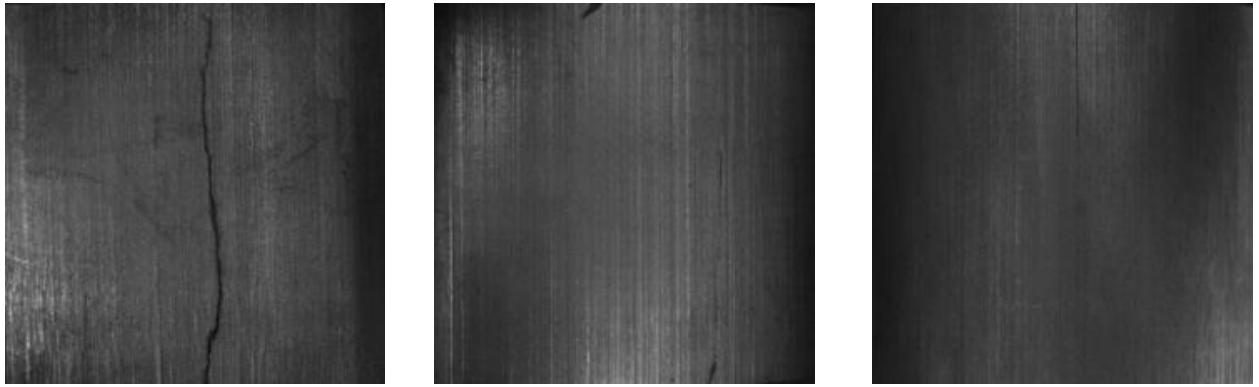
Unmasking Clever Hans Predictors, Pascal VOC Challenge

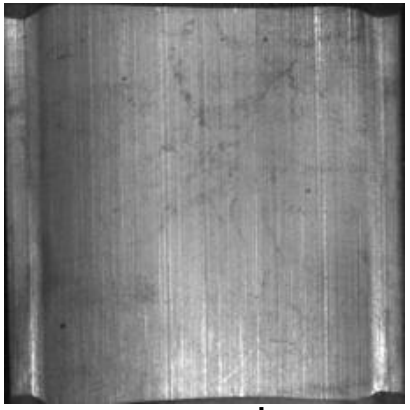


[Samek]

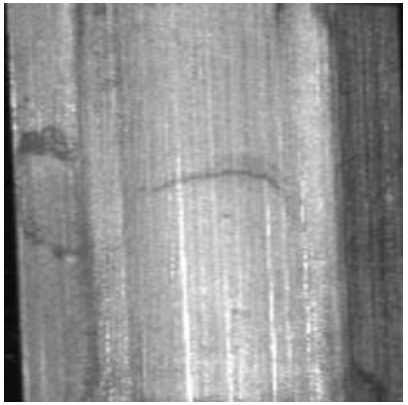
Material Defect Classification

surfaces with cracks





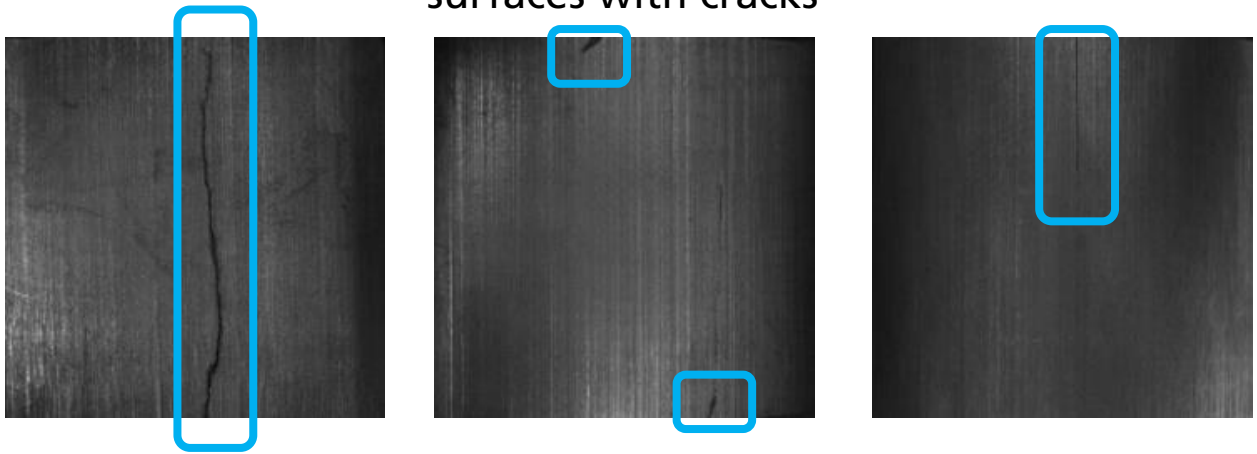
no cracks

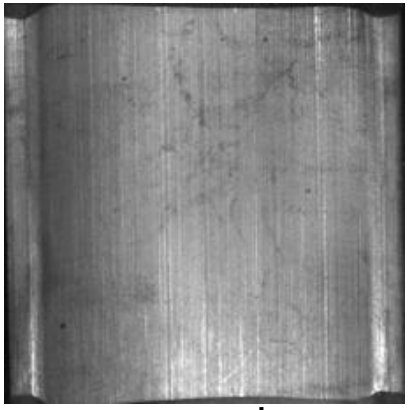


- Public dataset “magnetic tile surface defects”
- Network trained with 50 images with defects + 600 without
- Trained with labeled images without segmentation
- LRP for localization of defect

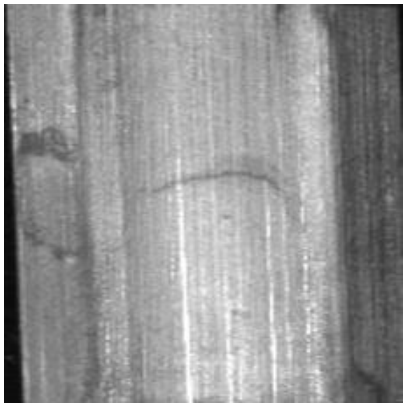
Material Defect Classification

surfaces with cracks



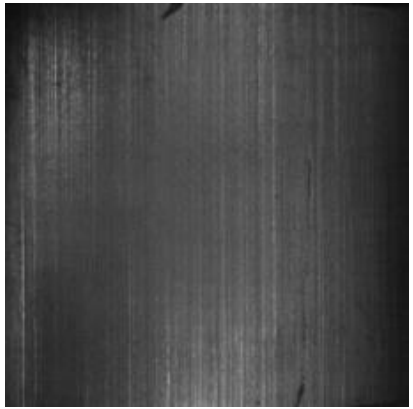


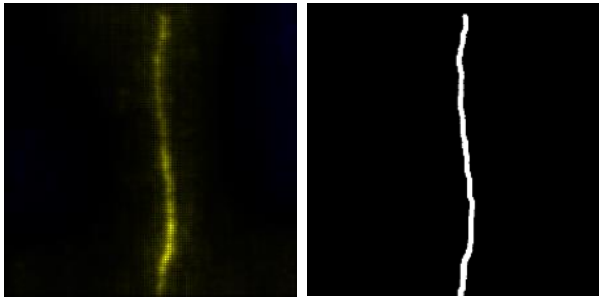
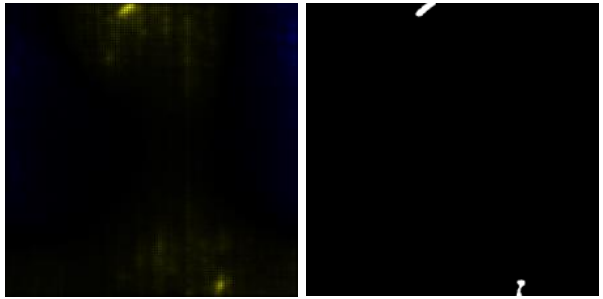
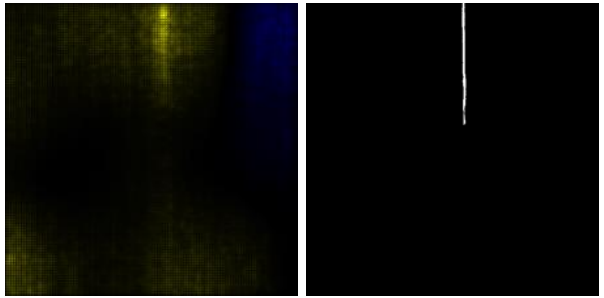
no cracks



- Public dataset “magnetic tile surface defects”
- Network trained with 50 images with defects + 600 without
- Trained with labeled images without segmentation
- LRP for localization of defect

Pixel-accurate Detection without Accurate Labeling

Rain Streak Detection

Input

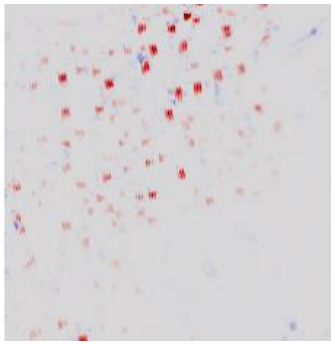
CNN Classifier

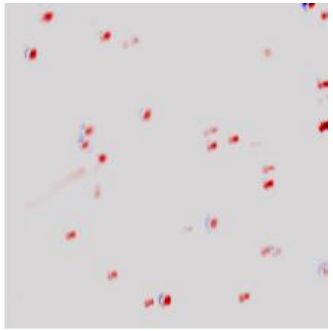
No (visible) rain

Rain

LRP

LRP highlights features of detected class -> should highlight the rain streaks





- LRP for detecting individual rain streaks
- Areas where computer vision methods may degrade
- No labeling of rain required

Classification of Face Images

female

glasses

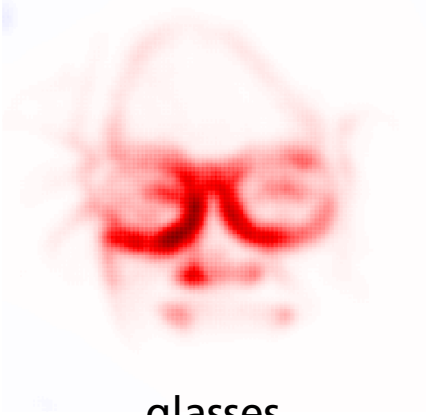
mustache

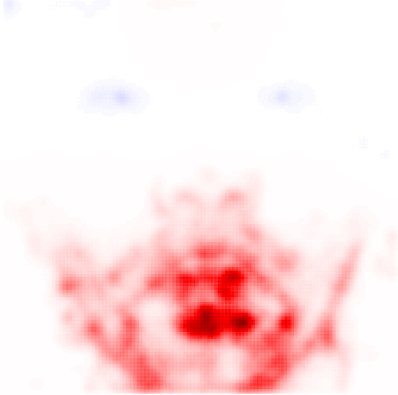
beard

- Deep Learning for classification of face images (gender, glasses, mustache, beard)
- Trained on about 20000 labeled face images (several public databases + google)

class	accuracy
gender	98.4%
glasses	99.9%
mustache	98.1%
beard	97.8%

Explanation Face Classification by LRP



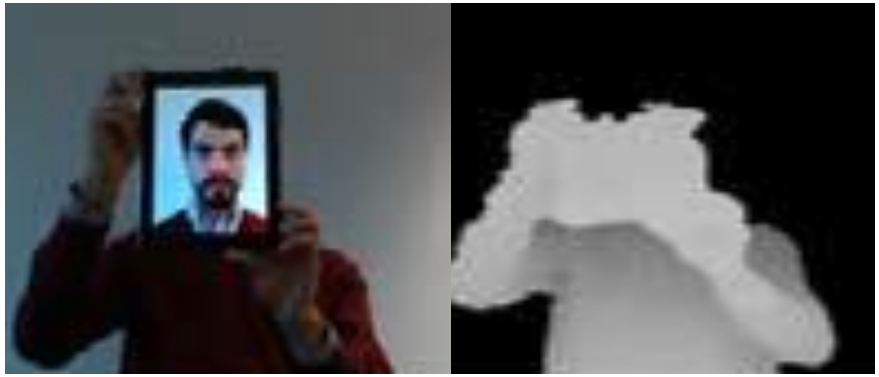


glasses

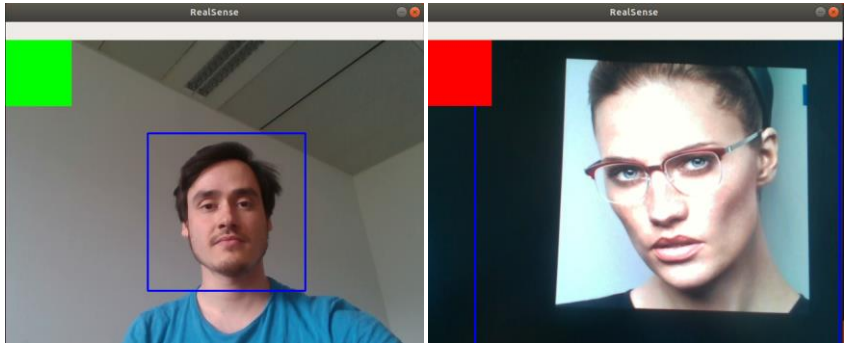
mustache

beard

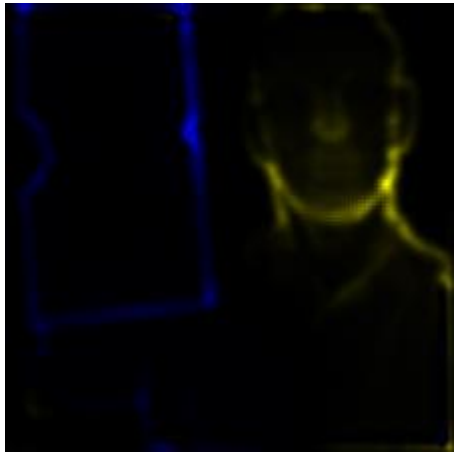
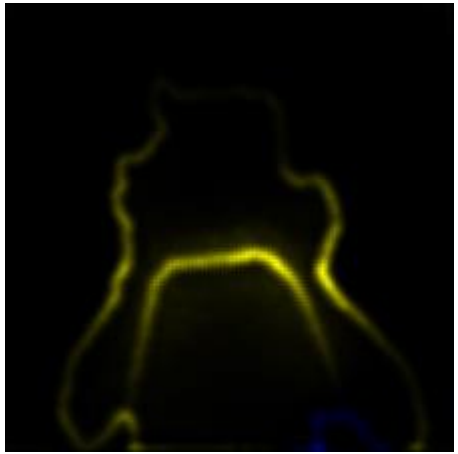
3D Face Geometry Classification



- Input: high quality 3D sensor & RealSense
- PAD: pictures, tablets
- Normalization of 3D face data
- Classification on depth maps
- Trained DNN (VGG19)
- 4 false positives from 631 samples

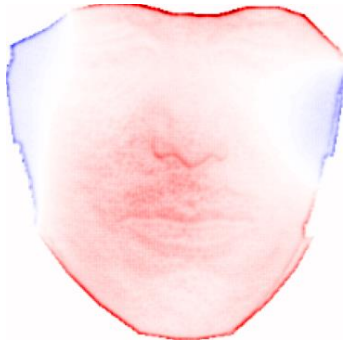
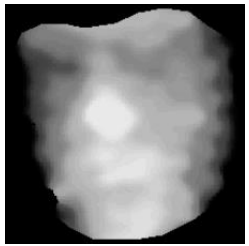
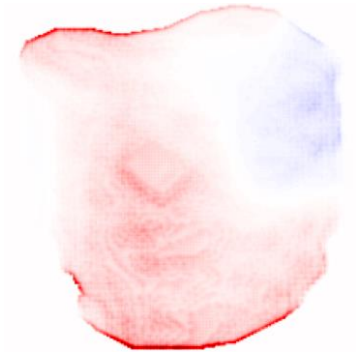


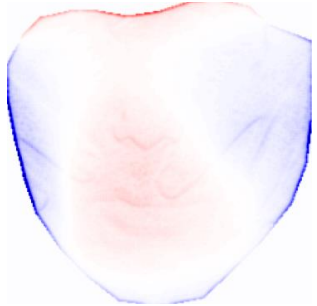
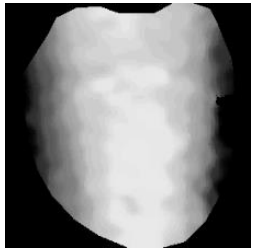
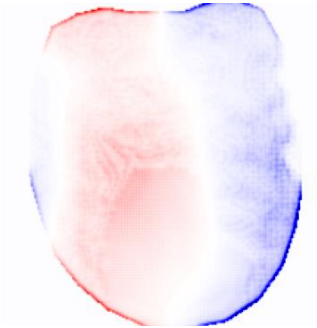
Explanation of 3D Face Geometry Classification (PAD)

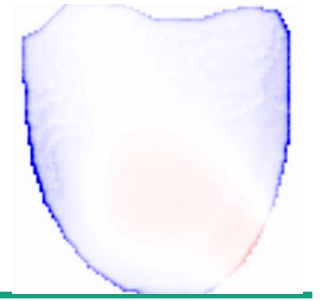



- Classifier mainly looked at the outline of the objects / person
- Learned the shape of the tablet rather than that of the face
- Prone to other attacks

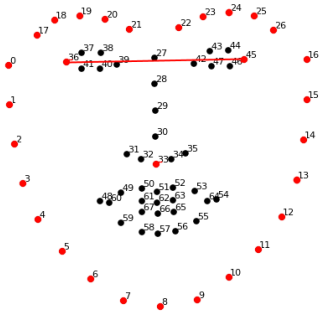
Enhanced 3D Face Geometry Classification

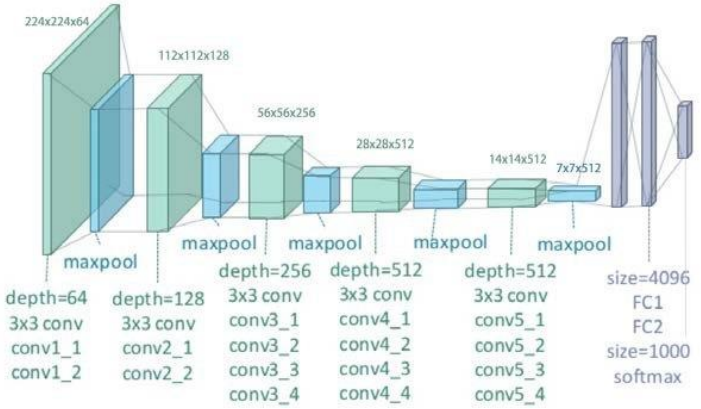






- Tracking and cutting out of face region
- Training and testing with face region only
- Network focuses on face geometry instead of outline



Detection of Face Morphing Attacks



- Morphing attack detection using Deep Neural Networks
- Creation of training and test datasets, >2000 original images
- Image pre-processing (filtering, noise) to increase variations in dataset
- Comparison of different network architectures, VGG19, GoogLe,…
- Equal error rate (EER): 3%

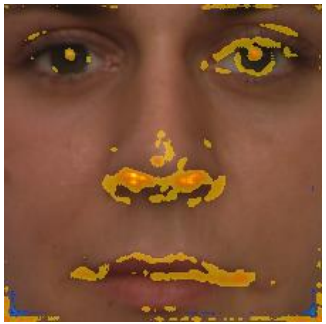
[Seibold, IWDW 2017]

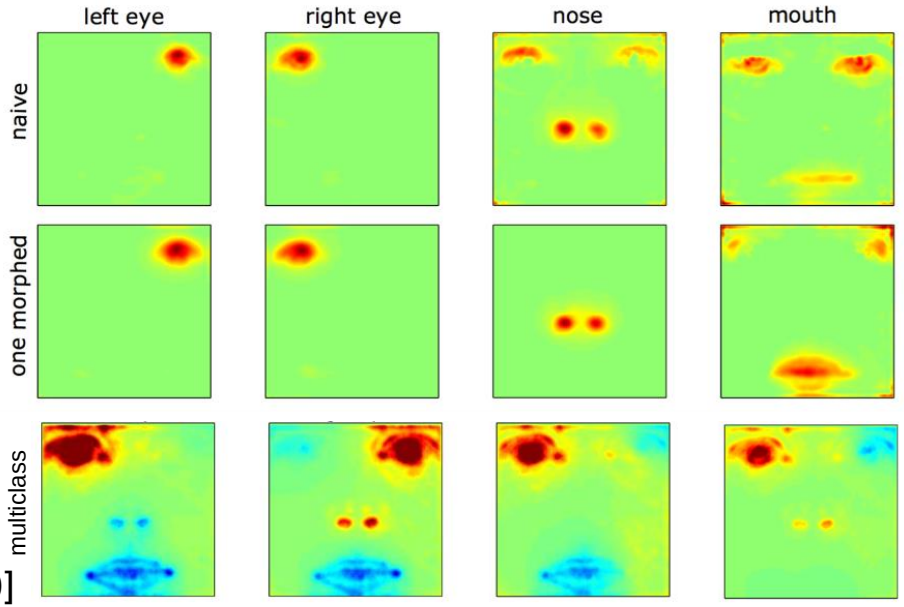
Visualization of Network Decisions for Face Morph Attacks

Enhanced Multiclass Training

- Multiclass network
- Enforce network to look at other regions as well
- Network has to detect which region is modified, retrained for full morph det.
- Networks learn different strategies

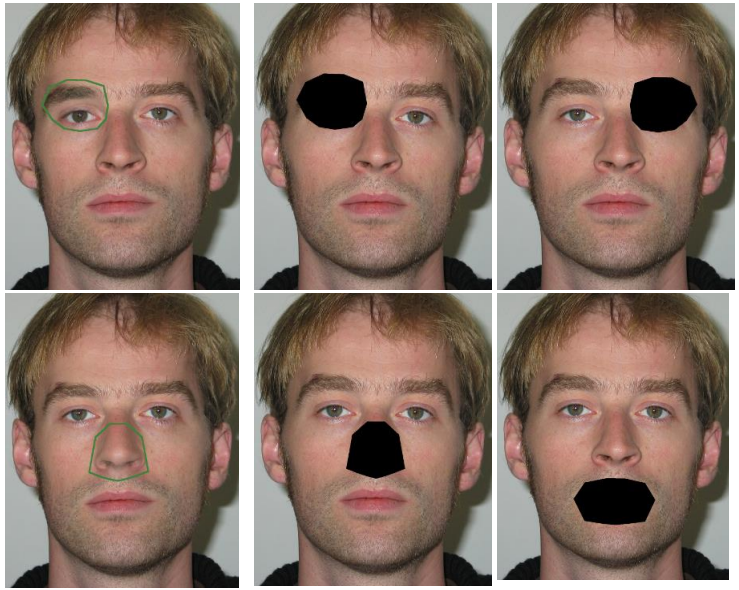
[Seibold, JISA 2020]

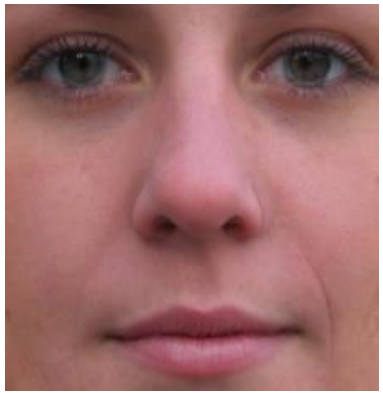




Enhanced Training

- Multiclass network
- Network has to detect which region is modified
- Much more robust to different attacks





simple face morph

fast gradient attack (±6)

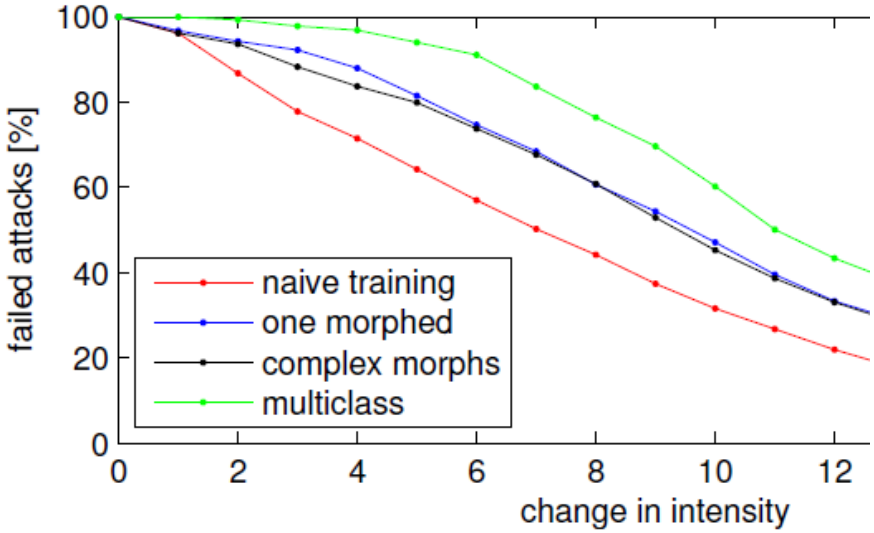
Training	EER	Partial morphs	Adver. attack
naive	3.1%	80%	53%
multi class	2.8%	13%	9%

Robustness against Fast Gradient Attacks

- Deception of classifier by minimal altering of images
- No knowledge about architecture / weights required

simple morph

fast gradient attack (ε= 3)



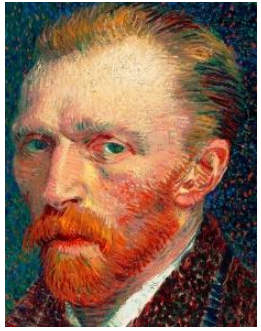
Neural Style Transfer (NST)

original image

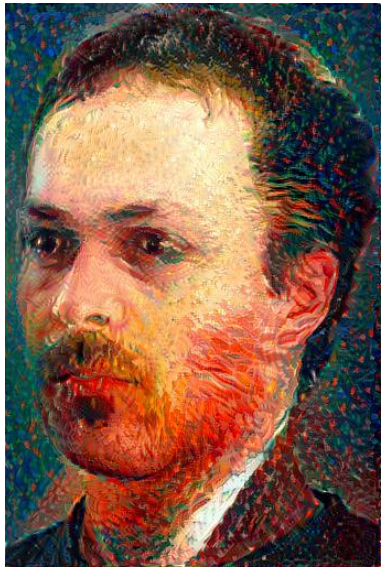
style image

[Gatys2015]

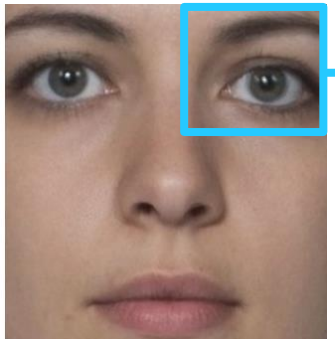
original image



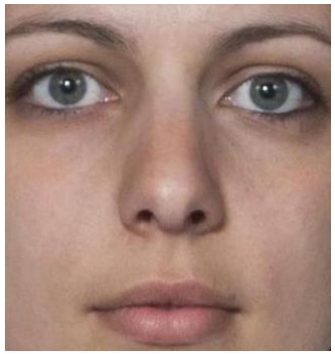
style image

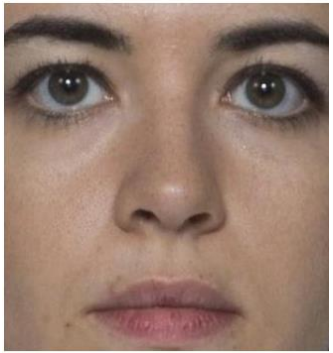


Neural Style Transfer (NST) for Image Enhancement

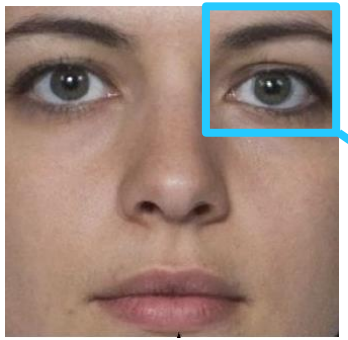


regular face morph





original images as style



enhanced morph

[Seibold, BIOSIG 2019]

Effects of Style Transfer for Morph Improvement

simple morph

NST improved morph

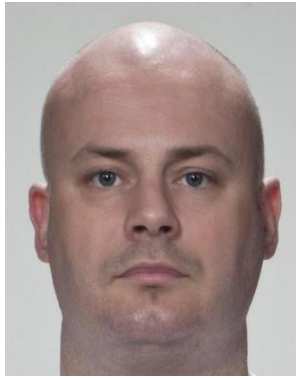
Effects of Style Transfer for Morph Improvement

simple morph

NST improved morph

Influence of Enhanced Morphs on Detection Rate

- Images from 9 different datasets
  - ~ 2,000 face images from different subjects
  - training on 50% genuine and 50% morphed face images
- Face morphing detectors evaluated
  - **Feature:** Edge Feature Based Detector [Krätzer17]
  - **LBP:** Local binary pattern [Raghavendra16]
  - **BSIF:** Binarized Statistical Image Features [Raghavendra16]
  - **DNN naïve:** Deep learning based detection [Seibold17]
  - **DNN MC:** multiclass DNN detection [Seibold18]




Different Image Enhancement Techniques

simple morph

simple

NST improved morph

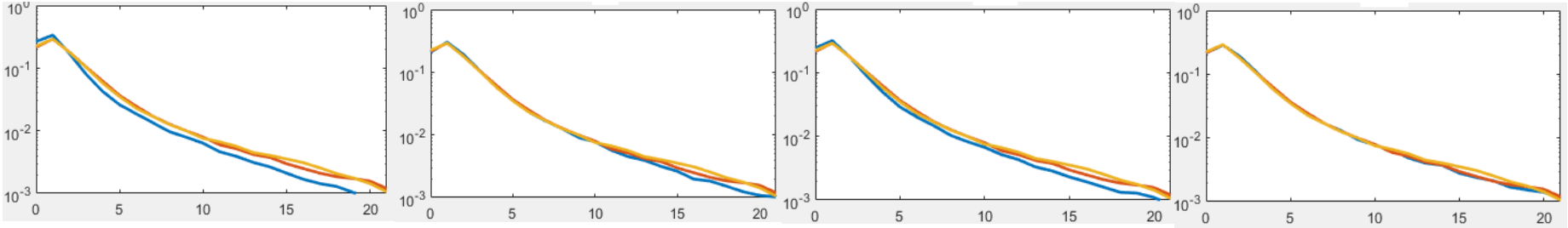
imp.

histogram equalization

HEQU

simple sharpening filter

sharp



gradient distribution (normalized):

Training only with Simple Morphs

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	32.6	17.3	54.6	43.6	49.7	74.7
LBP [RRB16]	25.4	21.1	60.3	58.5	35.1	65.7
BSIF [RRB16]	13.3	17.3	54.9	39.4	24.7	63.1
DNN naive [Se18]	1.5	1.0	30.7	3.1	32.5	72.6
DNN MC [Se18]	1.5	0.5	27.1	2.6	29.1	62.9

APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Training only with Simple Morphs

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	32.6	17.3	54.6	43.6	49.7	74.7
LBP [RRB16]	25.4	21.1	60.3	58.5	35.1	65.7
BSIF [RRB16]	13.3	17.3	54.9	39.4	24.7	63.1
DNN naive [Se18]	1.5	1.0	30.7	3.1	32.5	72.6
DNN MC [Se18]	1.5	0.5	27.1	2.6	29.1	62.9

APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Training only with Simple Morphs

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	32.6	17.3	54.6	43.6	49.7	74.7
LBP [RRB16]	25.4	21.1	60.3	58.5	35.1	65.7
BSIF [RRB16]	13.3	17.3	54.9	39.4	24.7	63.1
DNN naive [Se18]	1.5	1.0	30.7	3.1	32.5	72.6
DNN MC [Se18]	1.5	0.5	27.1	2.6	29.1	62.9

APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Training with Simple & NST Improved Morphs (50%/50%)

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	33.8	17.3	43.6	30.7	50.0	72.2
LBP [RRB16]	32.6	25.8	38.4	50.5	33.8	43.0
BSIF [RRB16]	17.4	10.6	31.7	34.8	19.6	38.7
DNN naive [Se18]	1.5	2.8	7.2	4.4	15.7	29.9
DNN MC [Se18]	1.8	1.8	3.4	2.1	9.3	6.4

APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Training with Simple & NST Improved Morphs (50%/50%)

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	33.8	17.3	43.6	30.7	50.0	72.2
LBP [RRB16]	32.6	25.8	38.4	50.5	33.8	43.0
BSIF [RRB16]	17.4	10.6	31.7	34.8	19.6	38.7
DNN naive [Se18]	1.5	2.8	7.2	4.4	15.7	29.9
DNN MC [Se18]	1.8	1.8	3.4	2.1	9.3	6.4

APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Training with Simple & NST Improved Morphs (50%/50%)

Detector	BPCER(%)	APCER(%) *simple*	*imp.*	*sharp*	*HEQU*	*imp.+HEQU*
Features [Kr17]	33.8	17.3	43.6	30.7	50.0	72.2
LBP [RRB16]	32.6	25.8	38.4	50.5	33.8	43.0
BSIF [RRB16]	17.4	10.6	31.7	34.8	19.6	38.7
DNN naive [Se18]	1.5	2.8	7.2	4.4	15.7	29.9
DNN MC [Se18]	1.8	1.8	3.4	2.1	9.3	6.4

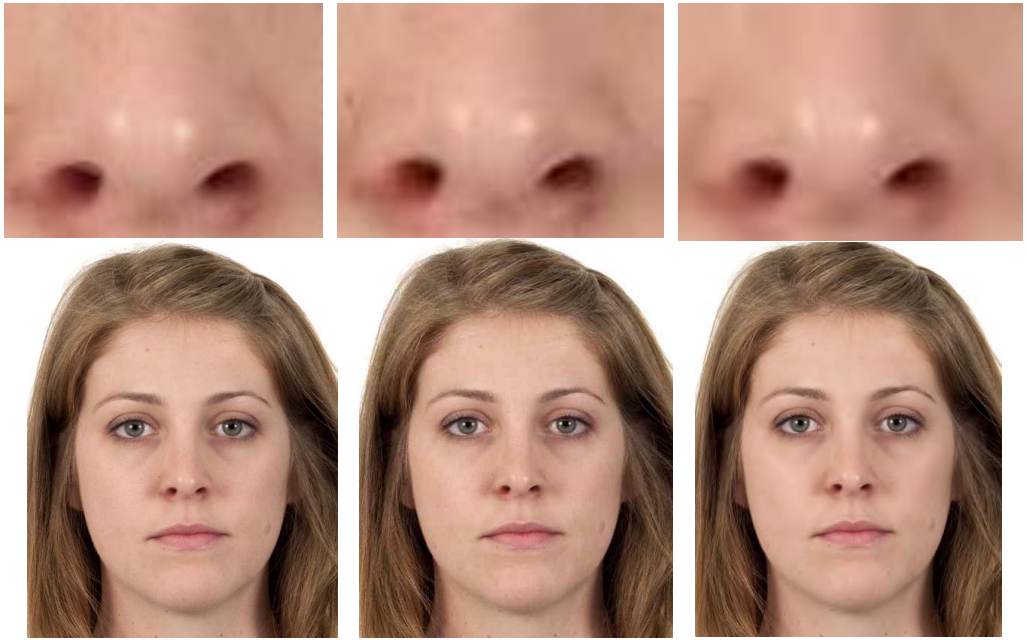
APCER - Attack Presentation Classification Error Rate (ISO/IEC 30107-3)
BPCER - Bona fide Presentation Classification Error Rate

Image Degeneration by Compression

- eMRTD - electronic Machine-Readable Travel Documents
  - image size: 413x531
  - JPEG or JP2000 compression
  - < 15360 bytes (15kB)

- How does this affect morphing attack detection systems?

[Seibold, WIFS 2019]



Performance and Generalization of the Detectors

- training with only uncompressed images

	Uncompressed			JP2000			JPEG		
	BPCER	APCER	EER	BPCER	APCER	EER	BPCER	APCER	EER
Benford	46.4%	6.5%	24.2%	3.8%	100.0%	50.0%	0.0%	100.0%	57.5%
Keypoint	24.7%	9.1%	17.7%	26.3%	39.2%	31.7%	30.1%	49.5%	38.2%
BSIF	14.0%	16.1%	15.6%	38.2%	8.6%	19.9%	19.4%	11.8%	17.2%
DNN	0.5%	4.3%	1.6%	1.1%	15.0%	7.0%	3.3%	4.8%	4.3%

BPCER: falsely as attack detected genuine face images
APCER: not detected morphed face images

Adaptability of the Detectors

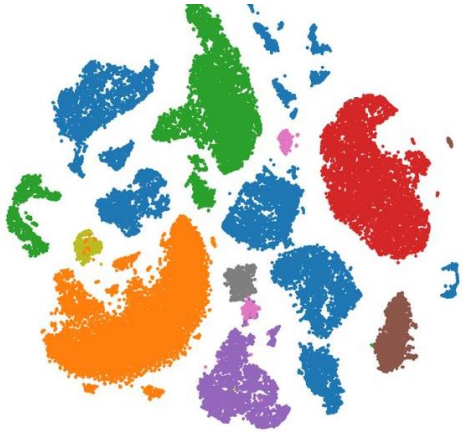
- training on corresponding compressed images

	Uncompressed			JP2000			JPEG		
	BPCER	APCER	EER	BPCER	APCER	EER	BPCER	APCER	EER
Benford	46.4%	6.5%	24.2%	18.8%	19.4%	18.4%	28.0%	18.2%	24.2%
Keypoint	24.7%	9.1%	17.7%	26.3%	15.6%	23.1%	22.6%	12.9%	17.2%
BSIF	14.0%	16.1%	15.6%	19.9%	19.4%	19.9%	12.4%	19.4%	15.0%
DNN	0.5%	4.3%	1.6%	6.5%	5.4%	5.9%	3.3%	4.8%	4.3%

BPCER: falsely as attack detected genuine face images
APCER: not detected morphed face images

Visualization of High Dimensional Features: t-SNE

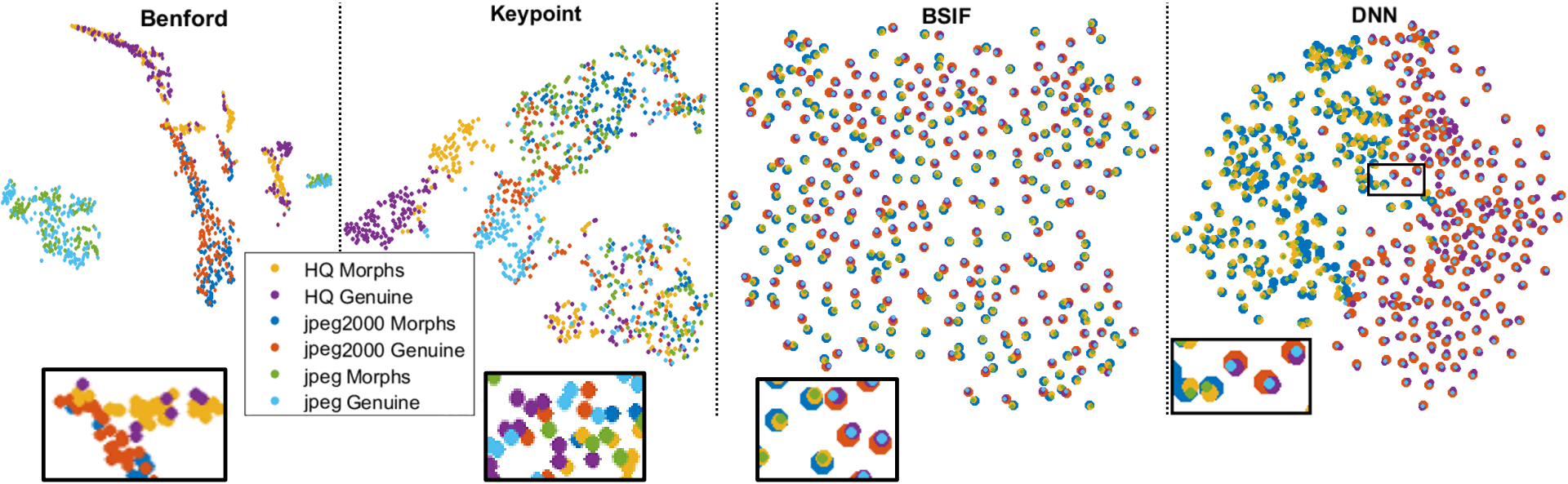
[van Maaten 2008]



- Visualization of high dimensional (HD) feature distribution in low (e.g. 2 or 3) dimensional space (LD)
- Features that are close in HD space are supposed to be close in LD space as well
  - Compute pairwise HD distance $p_{ij} \sim e^{-(x_i - x_j)^2 / 2\sigma^2}$
  - Normalize and make symmetric
  - Distribution of unknown LD features y: $q_{ij} \sim \left(1 + \left(y_i - y_j\right)^2\right)^{-1}$ (t-distribution)
  - Find y by minimizing KL distance $\sum_{i \neq j} p_{ij} \cdot log \frac{p_{ij}}{q_{ij}}_{ij}$ via gradient descent
  - Non-linear mapping, hyper parameters (perplexity (5-50), learning rate, steps) influence results

Distribution of all Features

- visualization using t-SNE



Explainability to Support Humans

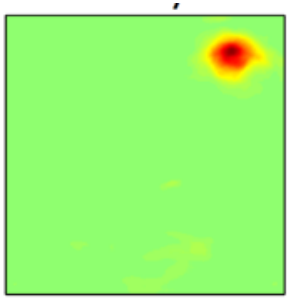
[Frankfurt Airport]


[Bundespolizei]

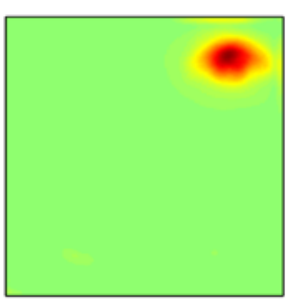
- LRP is great for optimizing classifiers
- A (technical) expert can learn how they work and why they came to a certain decision
- Support of border guards or enrolment officers?
  - If face verification fails, show which feature / face area is responsible
  - What is characteristic for a particular face
  - Show the artifacts if a face morphing attack is plausible

Average LRP for Partial Morphs

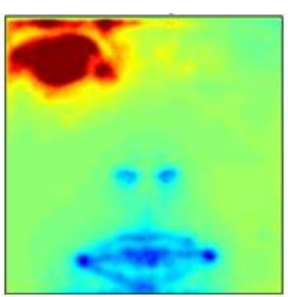
[Seibold, JISA 2020]



naive



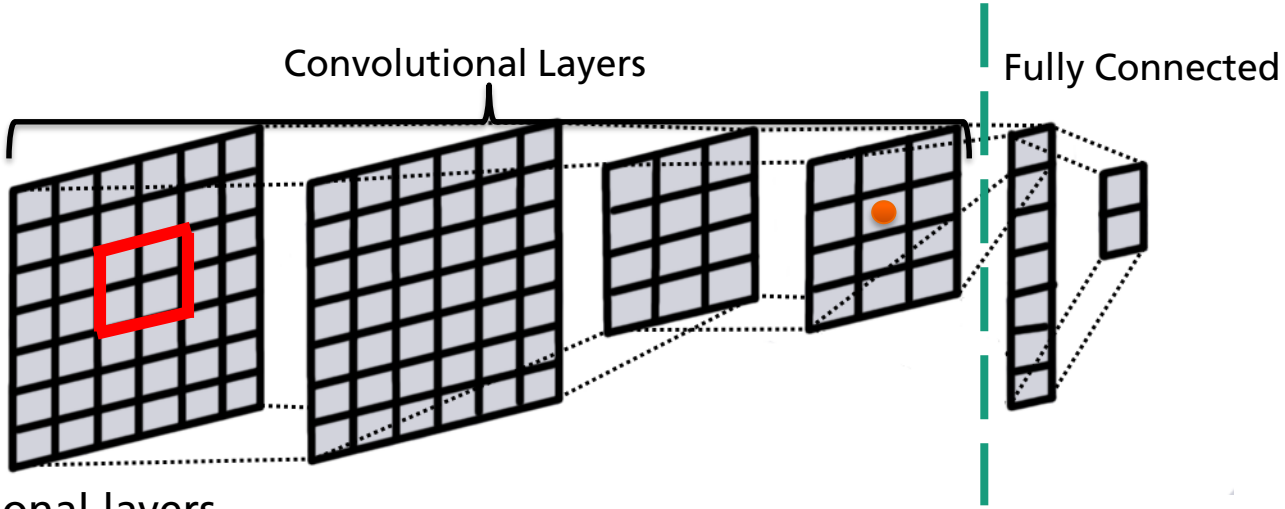
one morphed



multiclass

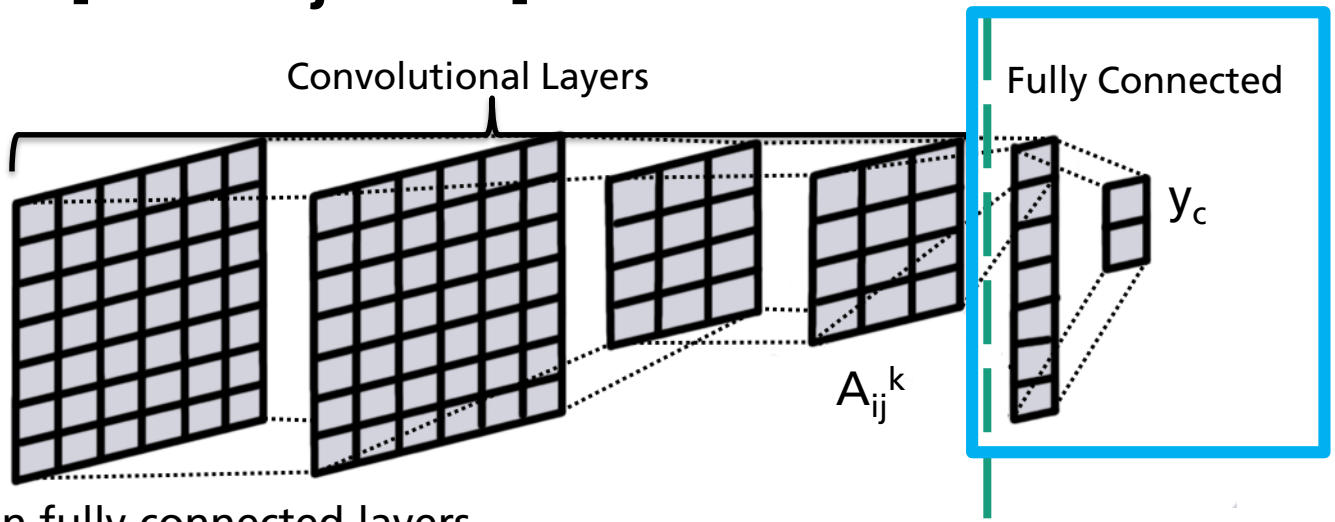
- For multiclass training, LRP seems to highlight the wrong eye
- Interpretation difficult because of complexity of fully connected layers (classifier)
- Here: the classifier learns to compare different image regions
- LRP highlights regions that lead to a strong activation of a neuron
- Difficult to interpret directly

Convolutional Neural Network



- Convolutional layers
  - Extract features via filtering the previous layer
  - Each feature in one layer corresponds to a fixed spatial region of the lower layer
- Fully connected layers
  - Act as classifier

Grad-CAM [Selvaraju2019]



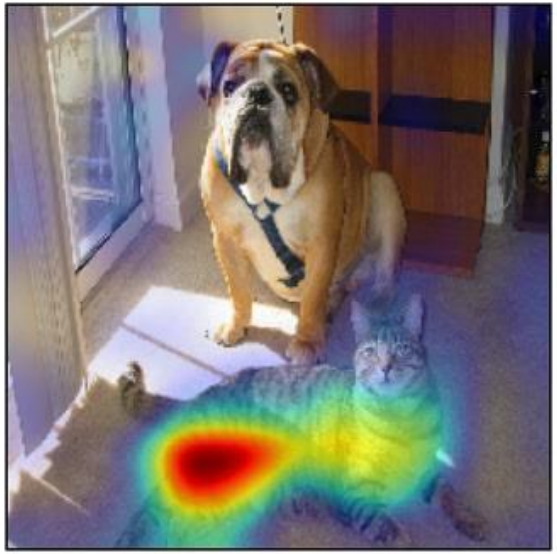
- Focuses on fully connected layers
- Complete forward pass, determine activation of highest conv layer A_{ij}^k and output y
- Compute importance of feature map A^k for decision y:
- Localization map L= $ReLU\left(\sum_k \alpha_k^c \cdot A^k\right)$
- Upsample for visualization (optionally multiplied with guided backprop)

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}}$$

Grad-CAM Example

[Selvaraju2019]

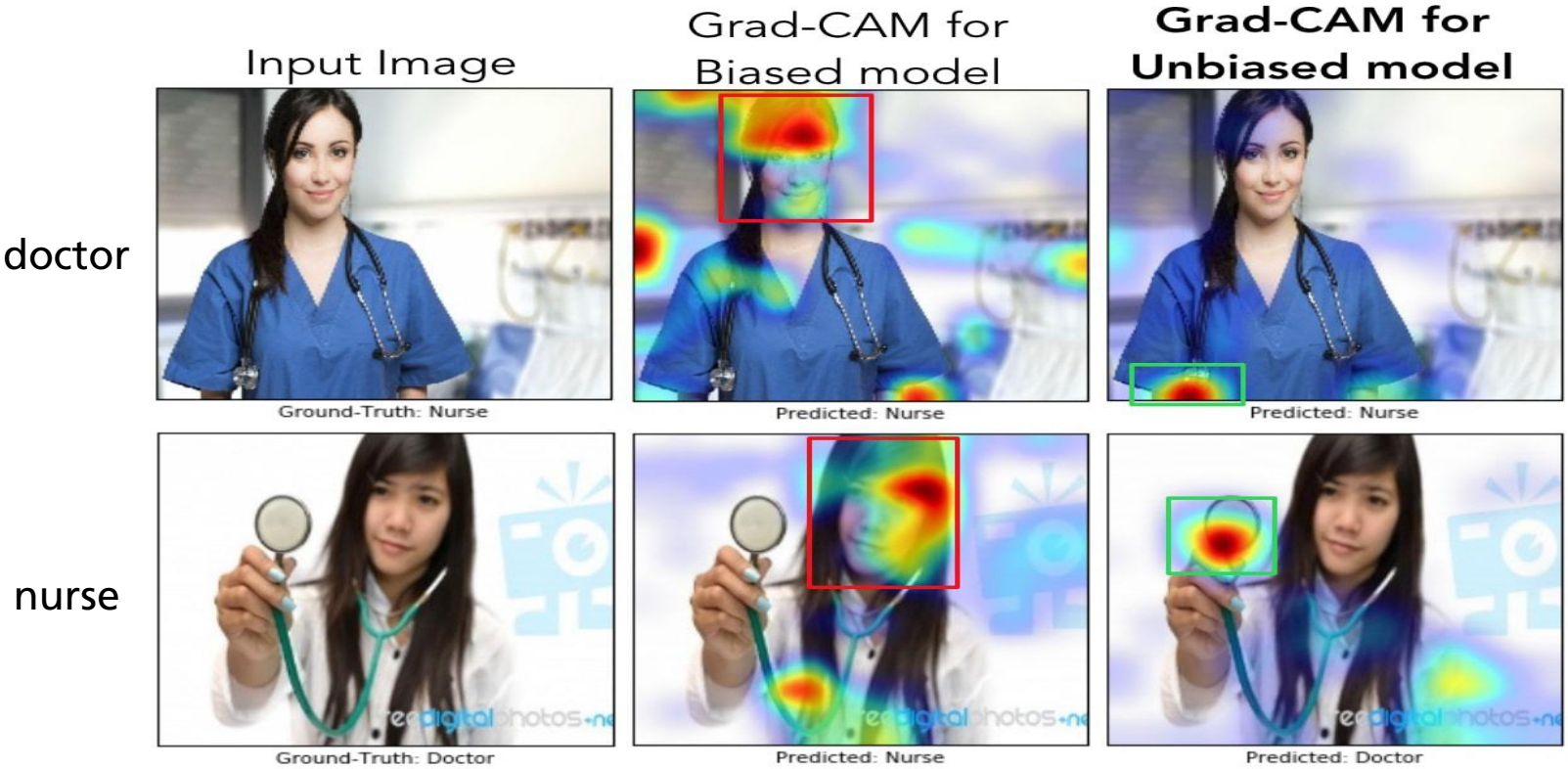
original



cat

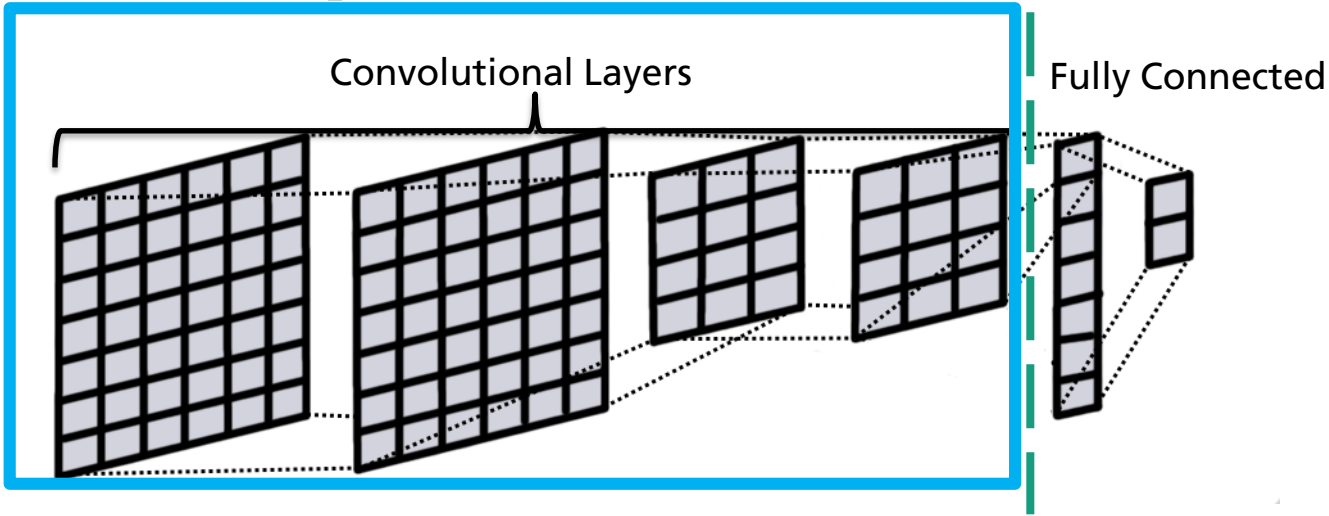
dog

Grad-CAM – Identifying Dataset Bias



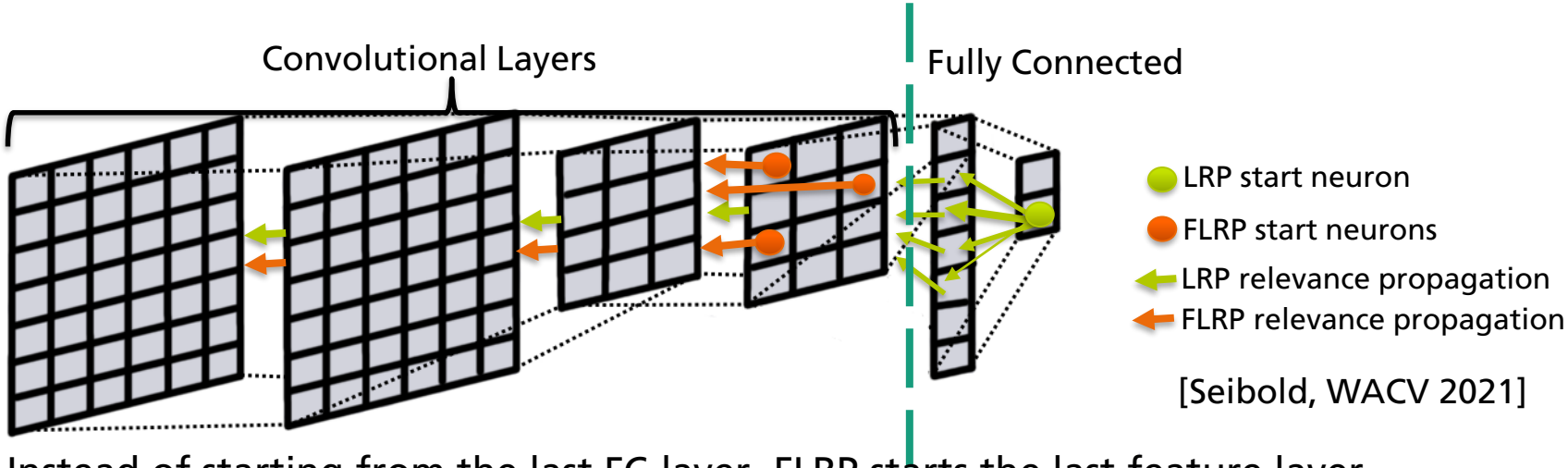
[Selvaraju2019]

FLRP [Seibold2021]



- Focuses on convolutional layers
- Use LRP for backpropagation of relevance (only conv layers)
- Spatial influence is fully preserved
- Initialization of relevance at highest conv layer required

FLRP: Focused Layer-wise Relevance Propagation

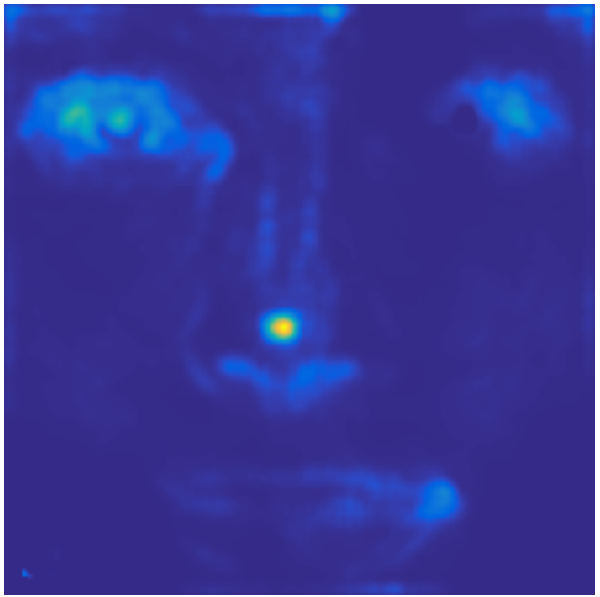


[Seibold, WACV 2021]

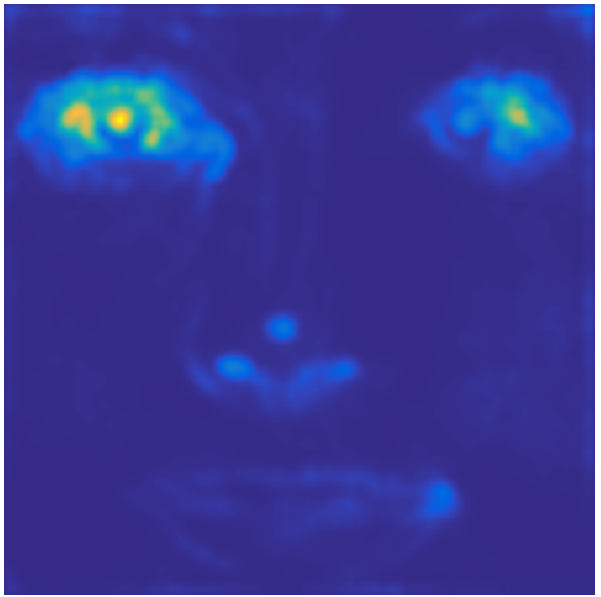
- Instead of starting from the last FC-layer, FLRP starts the last feature layer
- For each “pixel” neuron with best separating genuine and morphed images is selected
- Results similar to LRP but
  - FLRP highlights artifacts better
  - shows artifacts even if classifier is uncertain or even incorrect

Experiments – Example for Similar FLRP and LRP Results

input image



LRP

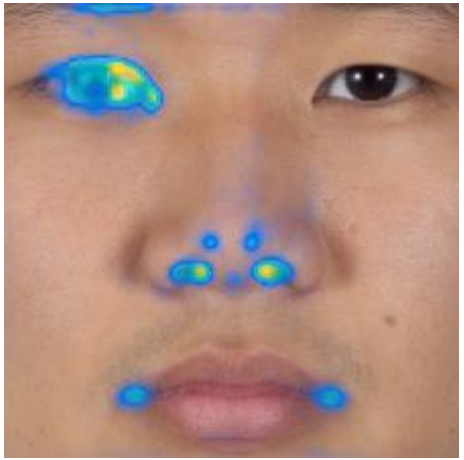


FLRP

LRP + FLRP for Undetected Face Morph

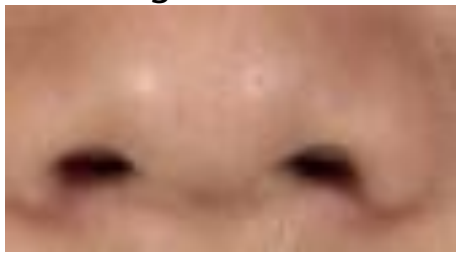
- FLRP mostly differs from LRP for naïvely trained MAD when the DNN's decision is wrong or uncertain
- In such cases:
  - LRP often highlights "good looking" regions
  - FLRP can still detect traces of forgery

FLRP



LRP (for class „Morph")

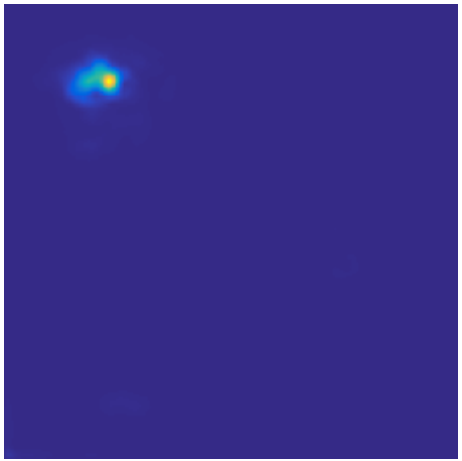
magnified nose



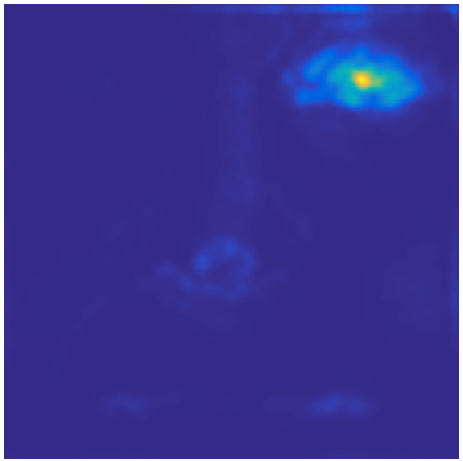
magnified eyes

LRP vs. FLRP

input image



LRP



FLRP

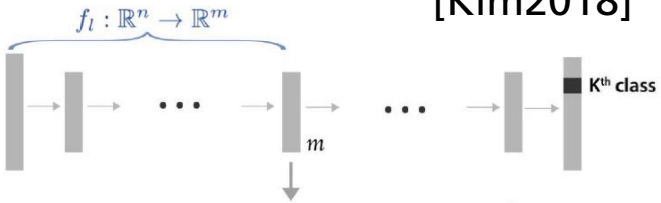
left eye

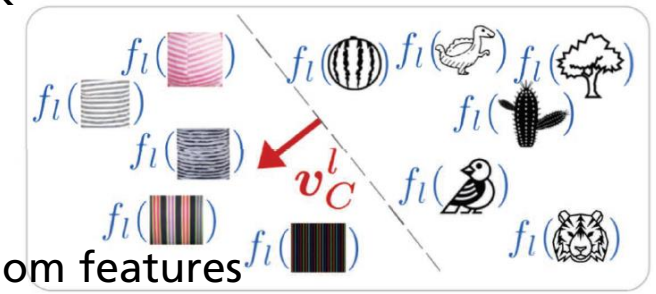
right eye

[Seibold 2021]

Testing with Concept Activation Vectors (TCAV)

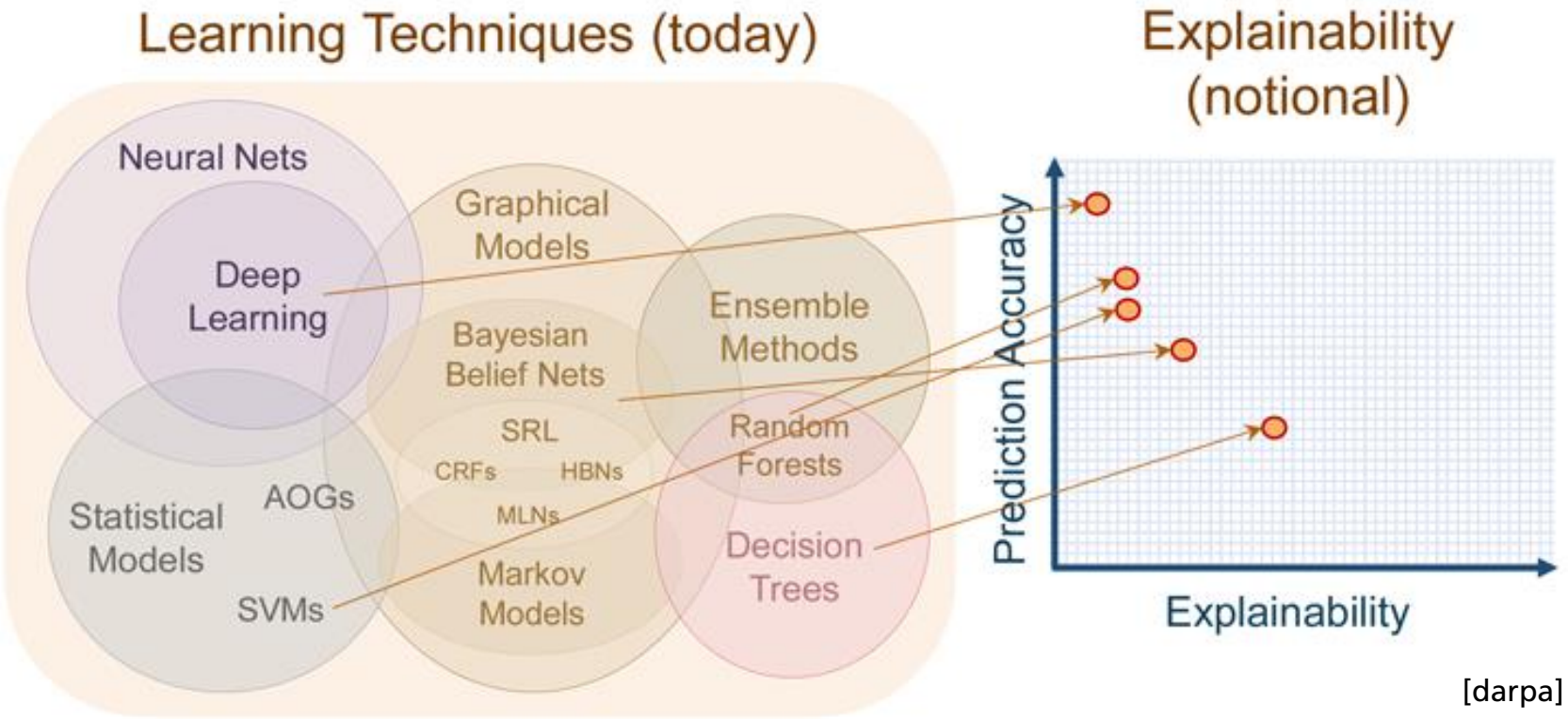
[Kim2018]





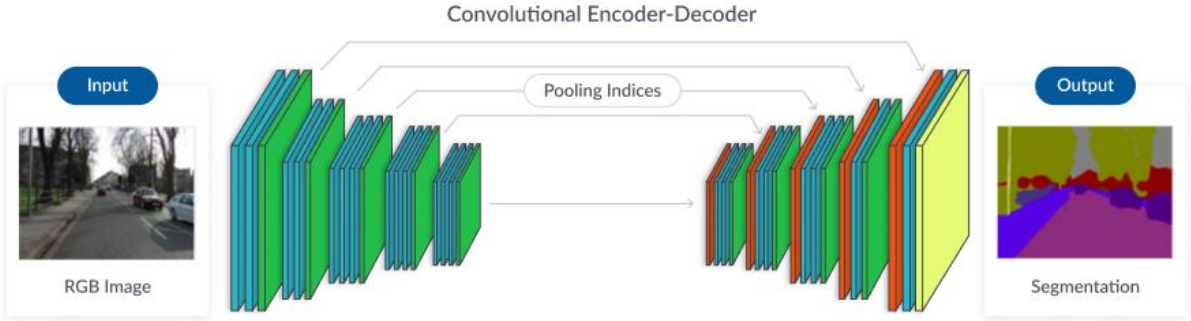
- Test whether a particular semantic concept (stripes for zebras) is important for the network
- Create 2 new datasets: one with images of the concept (stripes), the other ones with random images
- Inference of new test data, analyze bottleneck layer m
- Train a simple (linear) classifier separating concept/random features
- Compute sensitivity (derivative) of class K (for a given zebra image) with respect to a feature direction perpendicular to the separating line
- Provides information on how important this concept is for the particular output
- Rerun on different random datasets and for all images (zebras) in the testset (global explanation)

Simplifying Explainability Through Modified Architectures

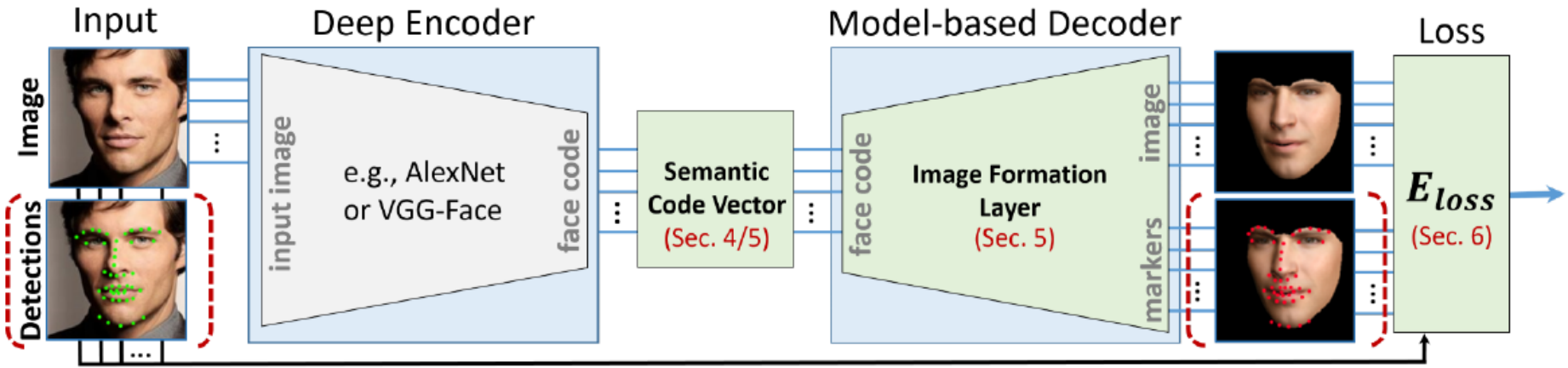


[darpa]

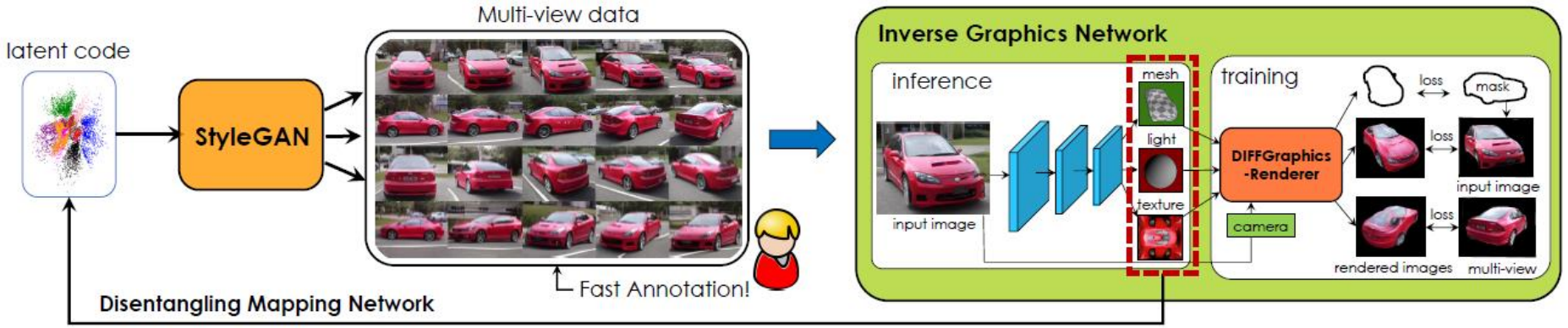
Autoencoder with Interpretable Latent Vector



[Tewari2017]



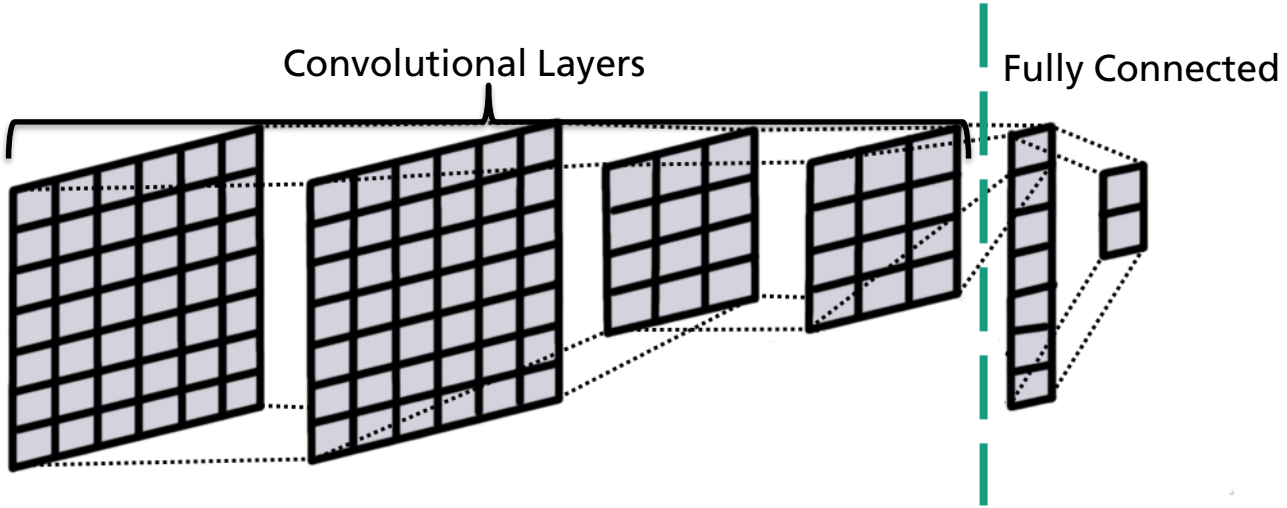
Interpretable Neural Rendering [Zhang 2021]



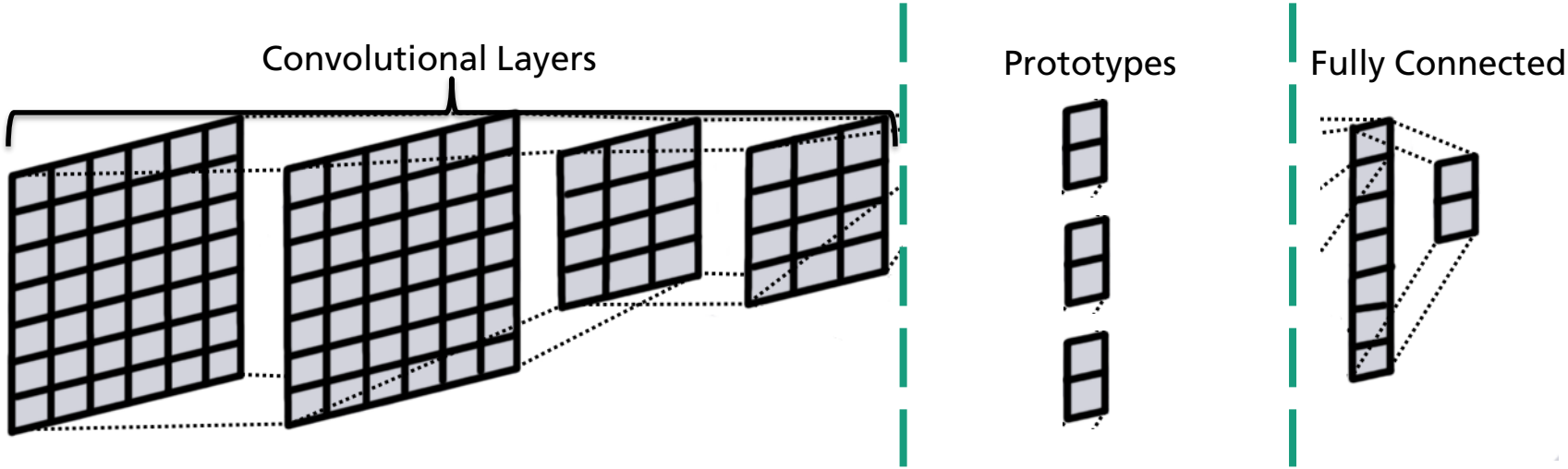
- Interpretability through explicit generation of sematic data (geometry, light, textures)
- Differential renderer for end to end learning

- Many new approaches with explicit data (physical properties, geometry etc.)

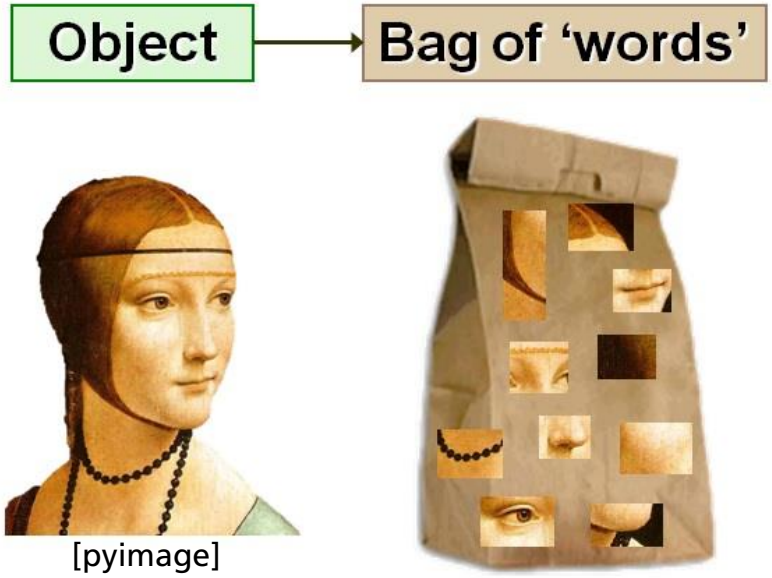
CNN Extension with Prototypes [Chen2019]



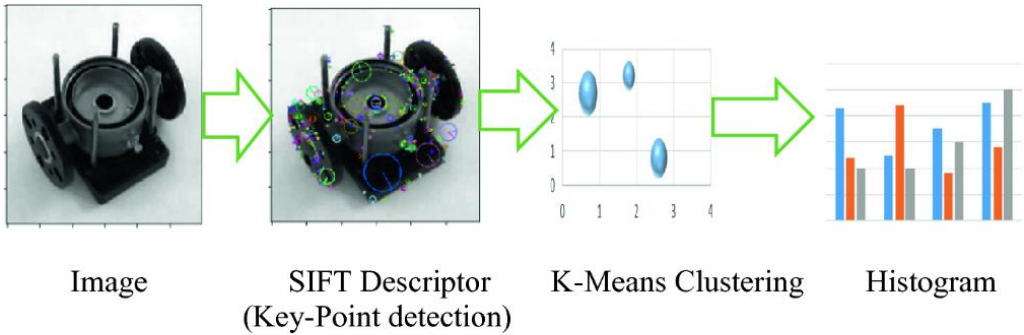
CNN Extension with Prototypes [Chen2019]



Object Classification using Bag of Visual Words



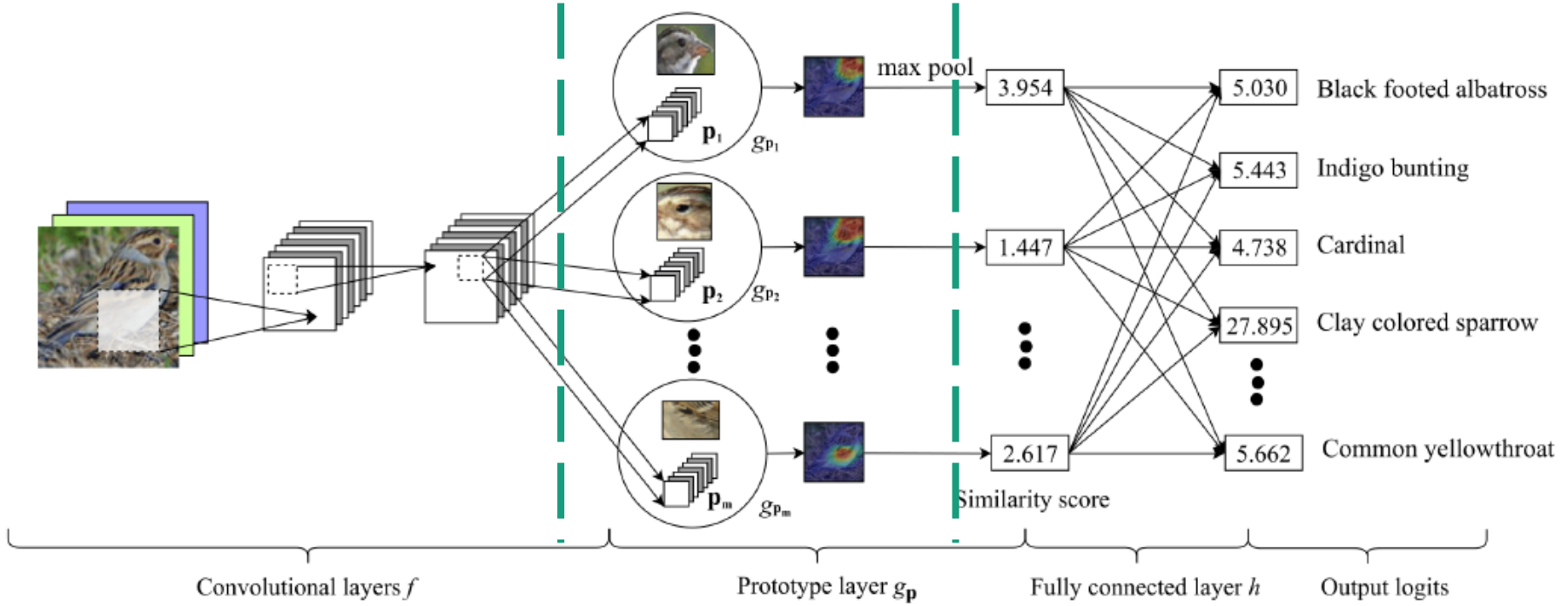
[pyimage]



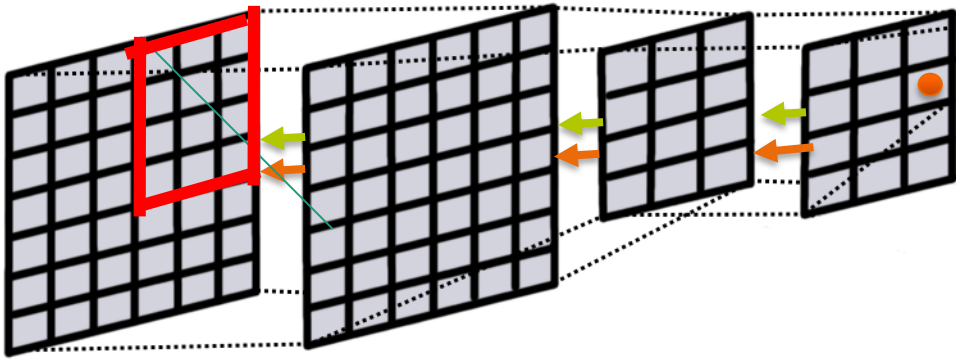
- SIFT feature describes small image region (with descriptor)
- Features are clustered, and cluster histogram are computed for an image
- Image assigned to class with similar cluster (prototype) histogram

ProtoPNet

[Chen2019: This looks like That]

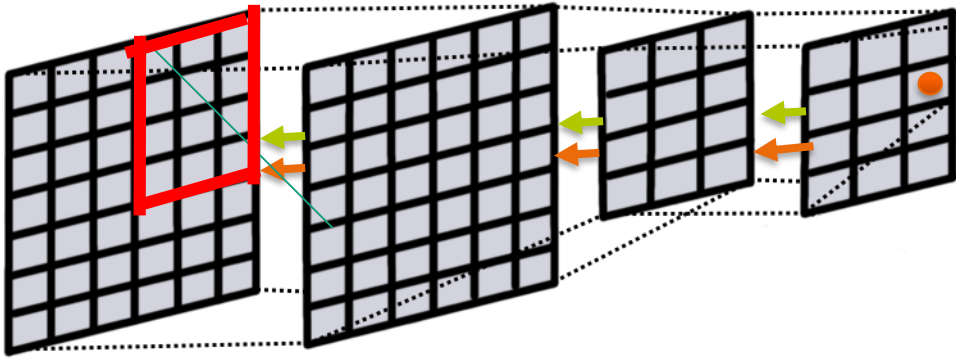


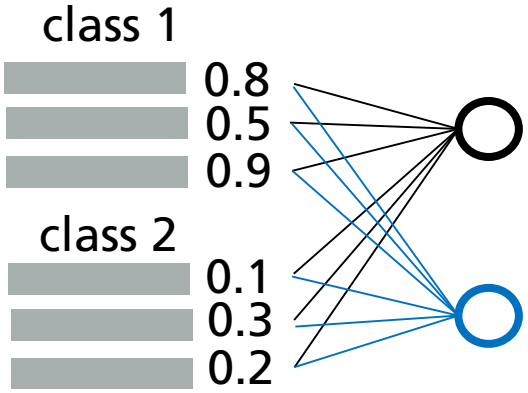
ProtoPNet

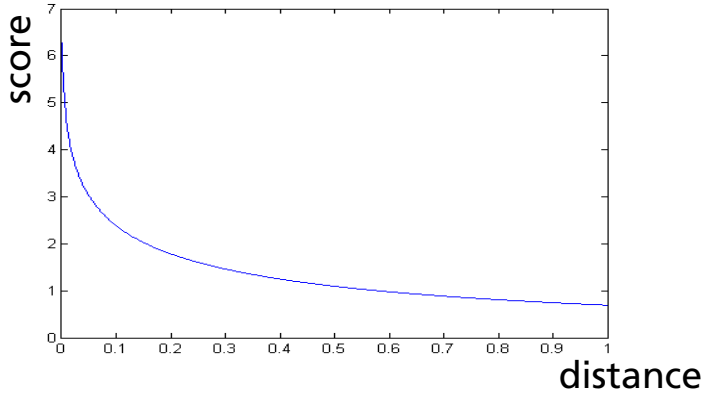


- Each pixel in feature map corresponds to a patch in the original image
- Describes the patch content with a 128, 256, 512 feature vector
- Each class stores N prototype vectors (characteristic for the class) corresponding to 1 pixel of the feature map (and a patch in the image)

ProtoPNet Inference

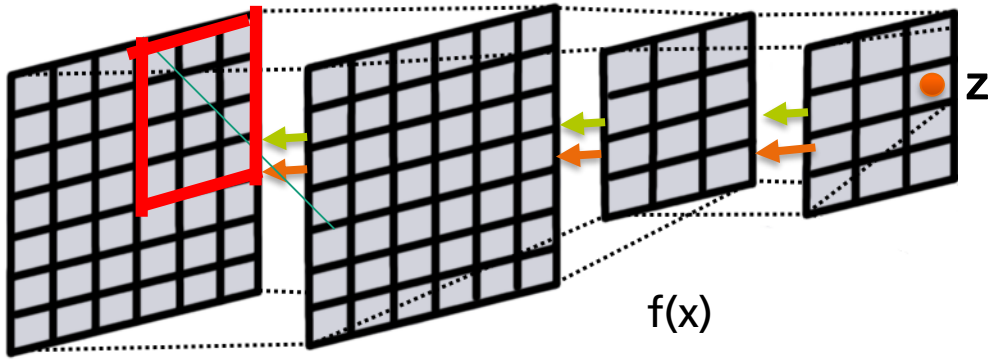


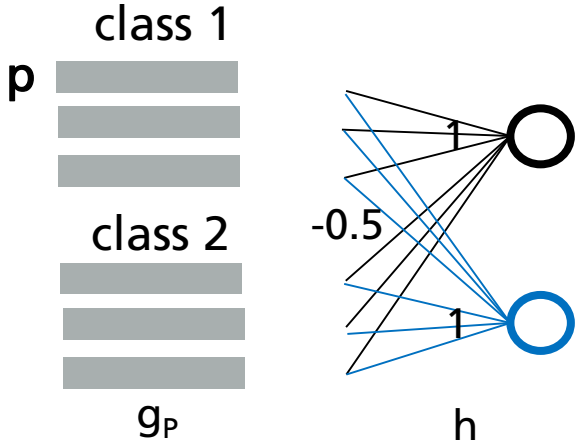




- Each prototype is compared with all vectors of the last feature map (is the prototype in the image?)
- Minimal distance L_2 (maximal similarity) is stored
- Matching score: max(log((d+1)/(d+ε))

ProtoNet Learning I





- First learn CNN + prototypes, stochastic gradient descent
- Fully connected layer fixed: w(correct class)=1, w(incorrect class)=-0.5

$$\min_{\mathbf{P}, w_{\text{conv}}} \frac{1}{n} \sum_{i=1}^{n} \text{CrsEnt}(h \circ g_{\mathbf{p}} \circ f(\mathbf{x_i}), \mathbf{y_i}) + \lambda_1 \text{Clst} + \lambda_2 \text{Sep},$$

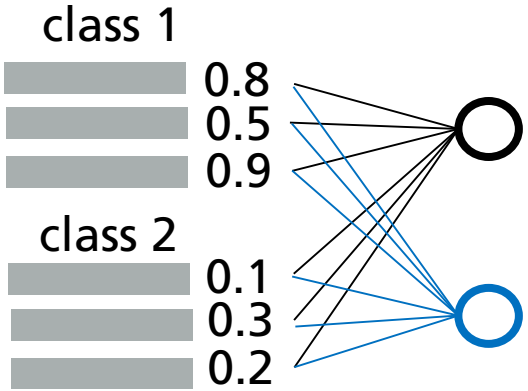
f: conv g: prototypes h: fully conv

$$\text{Clst} = \frac{1}{n} \sum_{i=1}^{n} \min_{j:\mathbf{p}_j \in \mathbf{P}_{y_i}} \min_{\mathbf{z} \in \text{patches}(f(\mathbf{x}_i))} \|\mathbf{z} - \mathbf{p}_j\|_2^2; \text{Sep} = -\frac{1}{n} \sum_{i=1}^{n} \min_{j:\mathbf{p}_j \notin \mathbf{P}_{y_i}} \min_{\mathbf{z} \in \text{patches}(f(\mathbf{x}_i))} \|\mathbf{z} - \mathbf{p}_j\|_2^2.$$

- Prototype projection: replace j-th prototype by the closest prototype from input data

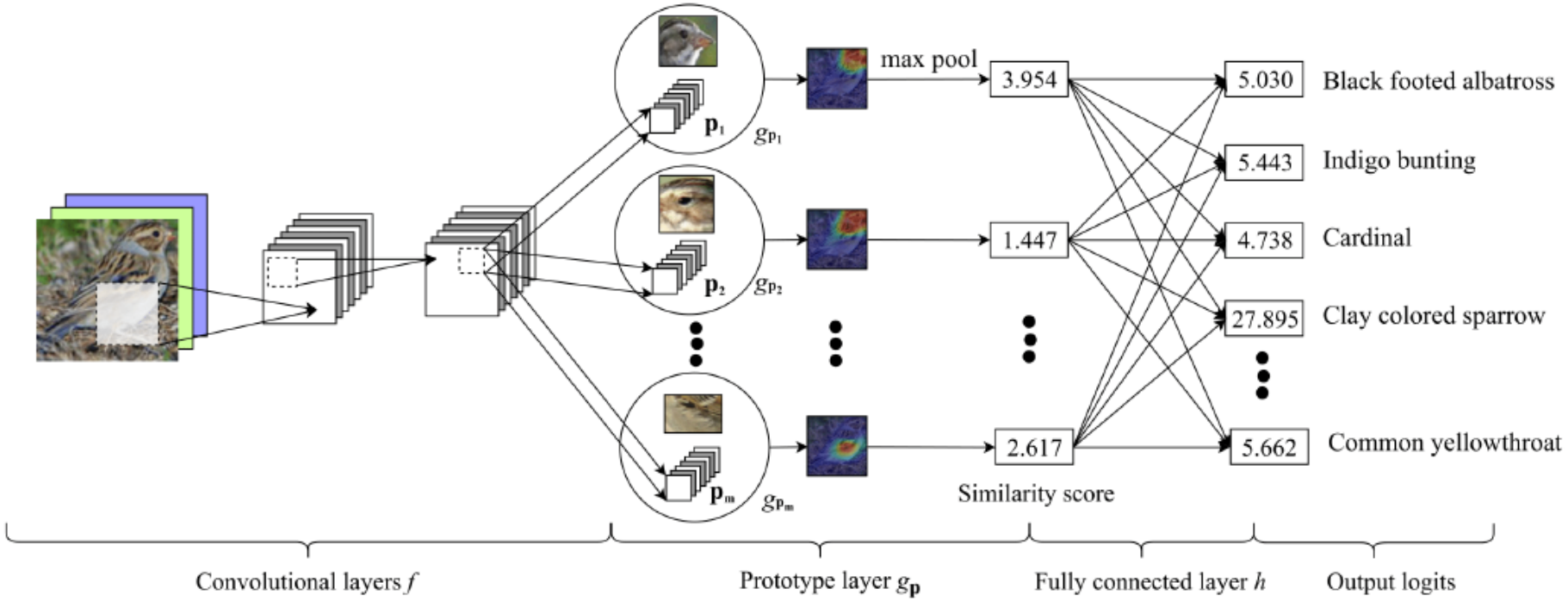
ProtoPNet Learning II

- Fix CNN + prototype
- compute weights of last convolutional layer
- convex optimization
- sparse results (negative prototype penalized)



$$\min_{w_h} \frac{1}{n} \sum_{i=1}^{n} \text{CrsEnt}(h \circ g_{\mathbf{p}} \circ f(\mathbf{x_i}), \mathbf{y_i}) + \lambda \sum_{k=1}^{K} \sum_{j:\mathbf{p}_j \notin \mathbf{P}_k} |w_h^{(k,j)}|$$

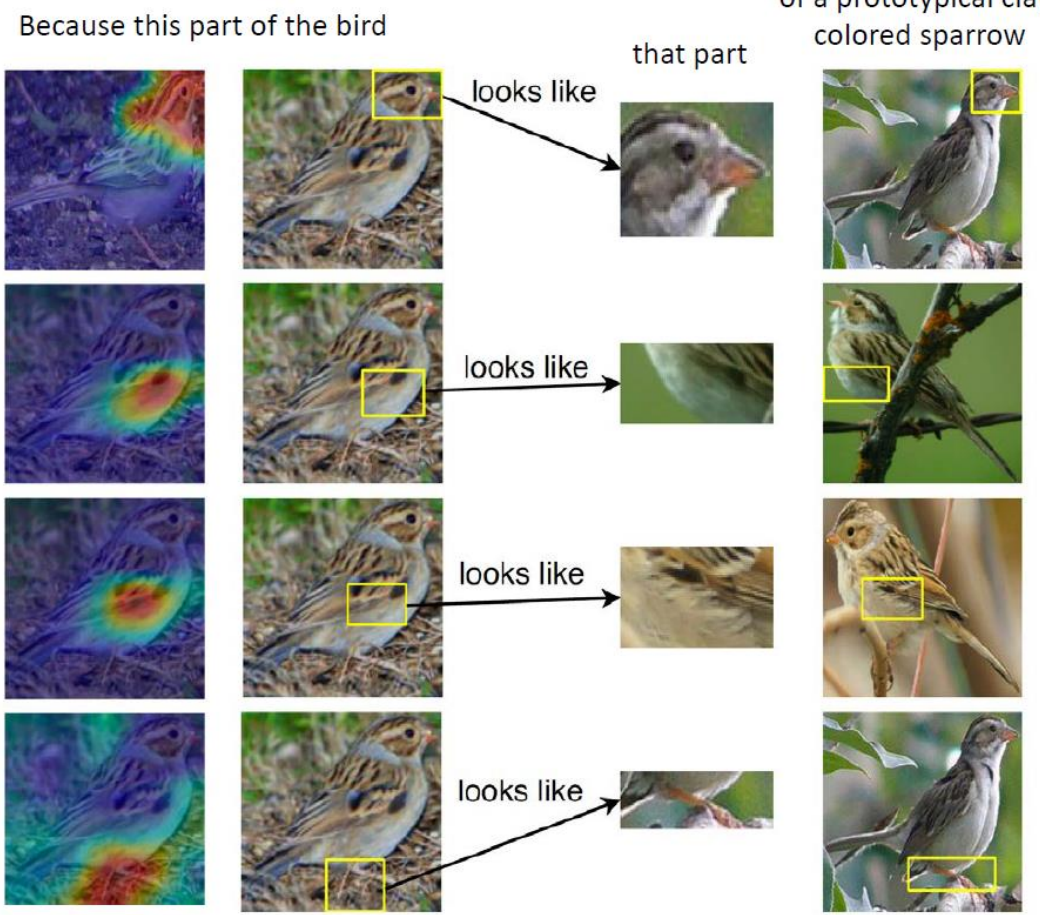
This looks like That



[Chen2019]

This looks like That

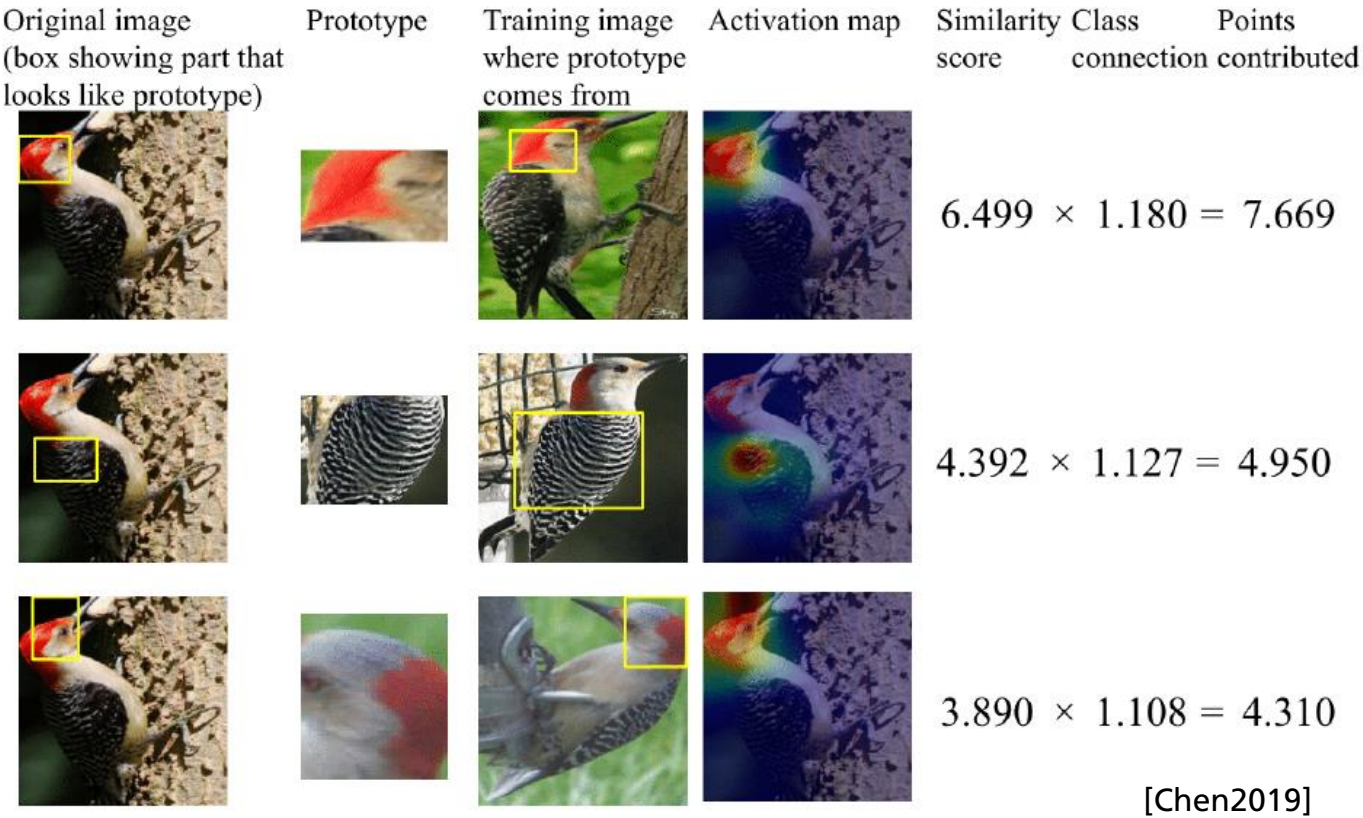
Why is this bird classified as a clay-colored sparrow?



[Chen2019]

Interpretability

red bellied woodpecker



[Chen2019]

Conclusions

- There are many potential threats on biometric identity verification systems
  - Deep neural networks provide promising solutions for their detection
  - Generalization is crucial to be robust against future attacks
  - Training database should be as diverse as possible
- Explainable AI is important for robust solutions
  - Black box systems are a problem for security/safety related applications
  - Many tools exist for explaining a network's decision and increase robustness (different results, might be misleading)
  - Human interpretability important (end user not only technical expert)
  - How to extend explainability to high dimensional signals (e.g. video)
  - Integration of classical models, prior knowledge can increase sematic interpretation

Thanks to all the Collaborators

- Aleixo Barreiro
- Philipp Fechteler
- Niklas Gard
- Anna Hilsmann
- Benjamin Kossack
- Johannes Künzel
- Wieland Morgenstern
- Wolfgang Paier
- Wojciech Samek
- Clemens Seibold
- Eric Wisotzky

Fraunhofer-Institut für Nachrichtentechnik, Heinrich-Hertz-Institut, HHI

WE PUT SCIENCE INTO ACTION.

Contact:

Peter Eisert
peter.eisert@hhi.fraunhofer.de
+49 30 31002 614

Einsteinufer 37
10587 Berlin

SPONSORED BY THE

Federal Ministry of Education and Research

Supported by:

Federal Ministry for Economic Affairs and Energy

on the basis of a decision by the German Bundestag

THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION
HORIZON 2020

References I

- C. Seibold, A. Hilsmann, P. Eisert, **Focused LRP: Explainable AI for Face Morphing Attack Detection**, *Proc. IEEE WACV Workshop on Explainabe AI for Biometry (xAI4biom)*, pp. 88-96, Jan. 2021
- C. Seibold, W. Samek, A. Hilsmann, P. Eisert, **Accurate and Robust Neural Networks for Face Morphing Attack Detection**, *Journal of Information Security and Applications*, 2020.
- C. Seibold, A. Hilsmann, A. Makrushin, C. Krätzer, T. Neubert, J. Dittmann, P. Eisert, **Visual Feature Space Analyses of Face Morphing Detectors to Predict Generalization Ability**, *Proc. IEEE Works. on Information Forensics and Security (WIFS 2019)*, Delft, Netherlands, Dec. 2019.
- B. Kossack, E. Wisotzky, A. Hilsmann, P. Eisert, **Local Remote Photoplethysmography Signal Analysis for Application in Presentation Attack Detection**, *Proc. Int. Workshop on Vision, Modeling, and Visualization*, Rostock, Germany, Oct. 2019.
- C. Seibold, A. Hilsmann, P. Eisert, **Style Your Face Morph and Improve Your Face Morphing Attack Detector**, *Proc. 18th Int. Conf. of the Biometrics Special Interest Group (BIOSIG 2019)*, Darmstadt, Germany, Sep. 2019.
- A. Makrushin, C. Kraetzer, J. Dittmann, C. Seibold, A. Hilsmann, P. Eisert, **Dempster-Shafer Theory for Fusing Face Morphing Detectors**, *Proc. European Signal Processing Conference (EUSIPCO)*, A Coruna, Spain, Sep. 2019.
- C. Seibold, A. Hilsmann, P. Eisert, **Reflection Analysis for Face Morphing Attack Detection**, *Proc. European Signal Processing Conference (EUSIPCO)*, Rome, Italy, Sep. 2018.
- C. Seibold, W. Samek, A. Hilsmann, P. Eisert, **Detection of Face Morphing Attacks by Deep Learning**, Proc. 16th Int. Workshop on Digital Forensics and Watermarking (IWDW), Aug. 2017.
- S Bach, A Binder, G Montavon, F Klauschen, KR Müller, W Samek. **On Pixel-wise Explanations for Non-Linear Classifier Decisions by Layer-wise Relevance Propagation**, PLOS ONE, 10(7):e0130140, 2015

References II

- C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, J. Su, **This Looks Like That: Deep Learning for Interpretable Image Recognition**, Proc. Advances in Neural Information Processing Systems 32 (NeurIPS), 2019.
- R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization,
- M. Sundararajan, A. Taly, Q. Yan, Axiomatic Attribution for Deep Networks, 2018. (Integrated Gradients)
- Y. Zhang, W. Chen, H. Ling, J. Gao, Y. Zhang, A. Torralba, S. Fidler, **Image GANs Meet Differentiable Rendering for Inverse Graphics and Interpretable 3D Neural Rendering**, *Proc. ICLR*, 2021.
- Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, Martin Wattenberg, **SmoothGrad: removing noise by adding noise,** 2017.
- A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Perez, C. Theobalt, **MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction**, *Proc. ICCV*, 2017.
- R. Raghavendra, K. B. Raja, C. Busch, **Detecting morphed face images**, *Proc. BTAS*, 2016.
- C. Krätzer, A. Makrushin, T. Neubert, M. Hildebrandt, and J. Dittmann, **Modeling attacks on photo-id documents and applying media forensics for the detection of facial morphing**, *Proc. IHMMSec*, 2017.
- M. Sharif, S. Bhagavatula, L. Bauer, M. Reiter, **Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition**, *Proc. ACM SIGSAC Conference on Computer and Communications Security*, Oct. 2016.
- S. Thys, W. Van Ranst, T. Goedemé, **Fooling automated surveillance cameras: adversarial patches to attack person detection**, *Proc. CVPRW*, 2019.

References III

- K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, D. Song, **Robust Physical-World Attacks on Deep Learning Visual Classification**, *Proc. CVPR*, 2018.
- L. van der Maaten, G. Hinton, **Visualizing Data using t-SNE,** *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.
- N. Carlini, D. Wagner, **Towards Evaluating the Robustness of Neural Networks**, *IEEE Symposium on Security and Privacy*, 2017.
- M. Ribeiro, S. Singh, C. Guestrin, **"Why Should I Trust You?" Explaining the Predictions of Any Classifier**, *Proc. International Conference on Knowledge Discovery and Data Mining (KDD)*, August 2016

Trustworthy AI Hands On Session

visum_trustworthy_ai_hands_on_session.ipynb ☆

File Edit View Insert Runtime Tools Help Changes will not be saved

+ Code + Text Copy to Drive

VISUM Summer School 2021 - Trustworthy AI Hands-on Session

Authors: *Clemens Seibold (Fraunhofer HHI), Johannes Künzel (Fraunhofer HHI), Peter Eisert (HU Berlin, Fraunhofer HHI)*

Objectives:

- Learn to implement and use LRP in PyTorch with different propagation rules
- Apply LRP to uncover undesired correlations in the data that are learnt by a Deep Convolutional Network (Clever-Hans effect)

This CoLab notebook consists of three parts: The implementation of three different LRP rules in PyTorch, an example that uncovers a Clever-Hans detector for gender classification and an exercise. During the exercise you can apply the described idea to uncover another Clever-Hans detector. To achieve this, you will have to select appropriate LRP rules and apply them on image data that we provide.

Import Python Packages and Download Necessary Resources

The necessary stuff to import packages and get example data... you can run this code without reading it.

```
[ ]  import torch
     import torch.nn as nn
     import torchvision.models as models

     from PIL import Image
     import numpy as np
     import copy
```

LeavesLRP.png

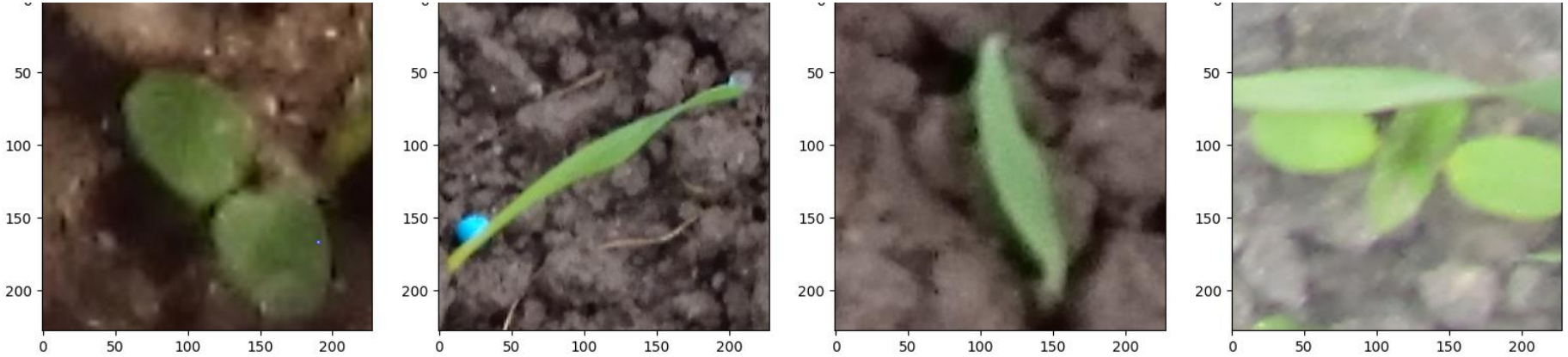
Johannes Künzel

Clemens Seibold

Gender Classification with Bias




Classify Number of Leafs (with some problems)



Fraunhofer-Institut für Nachrichtentechnik, Heinrich-Hertz-Institut, HHI

WE PUT SCIENCE INTO ACTION.

Contact:

Peter Eisert
peter.eisert@hhi.fraunhofer.de
+49 30 31002 614

Einsteinufer 37
10587 Berlin

SPONSORED BY THE

Federal Ministry of Education and Research

Supported by:

Federal Ministry for Economic Affairs and Energy

on the basis of a decision by the German Bundestag

THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION
HORIZON 2020