Eye Localization using the Discriminative Generalized Hough Transform

Ferdinand Hahmann, Dipl.-Inf.

Institute of Applied Computer Science
University of Applied Sciences Kiel

Graz – 30.08.2012

Eye Localization

1. Introduction
2. Method
   - Generalized Hough Transform (GHT)
   - Discriminative Generalized Hough Transform (DGHT)
   - Multi-Level-Approach
3. Experiments
   - Database
   - Setup
   - Results
   - Discussion
4. Conclusion

Introduction

- Motivation
  - Eye positions required by many face processing algorithms
- State-of-the-Art
  - Face detector developed by Viola & Jones
  - Usage of a-priori knowledge for eye detection inside faces
- The Discriminative Generalized Hough Transform (DGHT) [Sch07, Rup10] is a general object localization approach
  - Successfully applied in medical image analysis
  - Goal: Transfered to a new field of application

[Sch07]: H. Schramm, "Automatic 3-D object detection", Patent Pub. No. W0/2007/07/2391, 2007.
[Rup10] H. Ruppertshofen et al, "Iterative Training of Discriminative Models for the Generalized Hough Transform"', MICCAI, 2010

Outline

1 Introduction

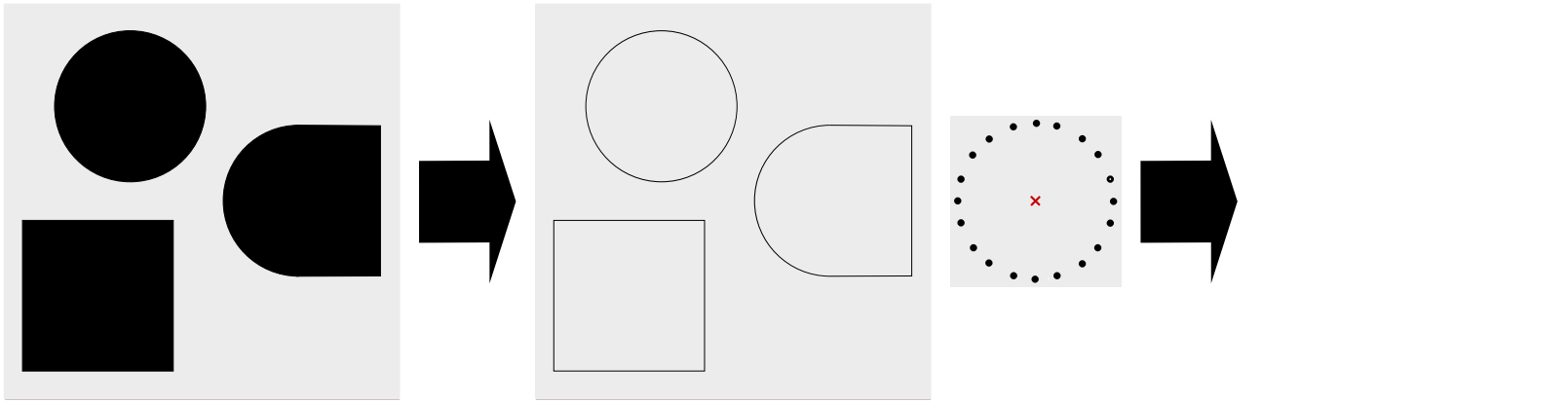
2 Method

3 Experiments

4 Conclusion

Generalized Hough Transform

- Introduced by Ballard 1981 [Bal81]
- General model-based method for object localization



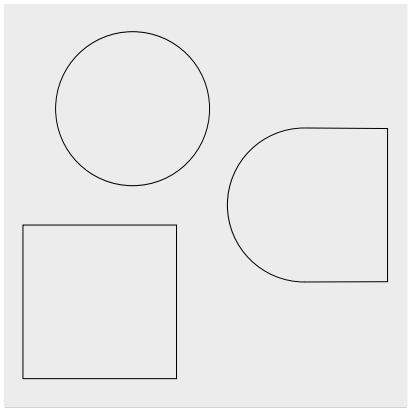
- Here: Application of edge features
- No model transformations considered

[Bal81]: D. Ballard, Generalizing the Hough transform to detect arbitrary shapes, Pattern Recogniton 13(2), 1981

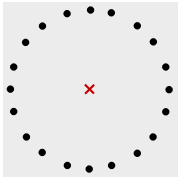
Discriminative Generalized Hough Transform (DGHT)

Motivation

- Discriminative GHT Model:

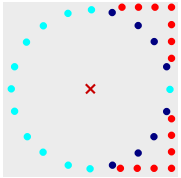


feature image



model

Hough space



weighted model

model point weights:
• = 0.5 • = 1 • =-1

weighted Hough space

- Model is composed of
  - geometric model layout
    $\rightarrow$ learned from training examples
  - individual weights of model points
    $\rightarrow$ discriminative weighting procedure

Model point weight estimation

- Generation of Hough spaces for training images, using initial GHT model
- Separation of Hough space votes coming from every single model point
- Weighted recombination of model point contributions into Maximum Entropy Distribution
- Optimization of introduced weights λ_j by using a Minimum Classification Error (MCE)

Model point weight estimation

- Generation of Hough spaces for training images, using initial GHT model
- Separation of Hough space votes coming from every single model point

$$f_j(c_i, X_n) = v_{i,j}$$

$v_{i,j}$: Votes of model point m_j in Hough cell c_i
X_n : Features of image n

- Weighted recombination of model point contributions into Maximum Entropy Distribution
- Optimization of introduced weights λ_j by using a Minimum Classification Error (MCE)

Model point weight estimation

- Generation of Hough spaces for training images, using initial GHT model
- Separation of Hough space votes coming from every single model point: $f_j(c_i, X_n)$
- Weighted recombination of model point contributions into Maximum Entropy Distribution

$$p_\Lambda(c_i|X_n) = \frac{\exp\left(\sum_j \lambda_j \cdot f_j(c_i, X_n)\right)}{\sum_k \exp\left(\sum_j \lambda_j \cdot f_j(c_k, X_n)\right)}$$

$\Lambda = [\lambda_1, \lambda_2, ..., \lambda_J]$: Individual model point weights

- Optimization of introduced weights λ_j by using a Minimum Classification Error (MCE)

Model point weight estimation

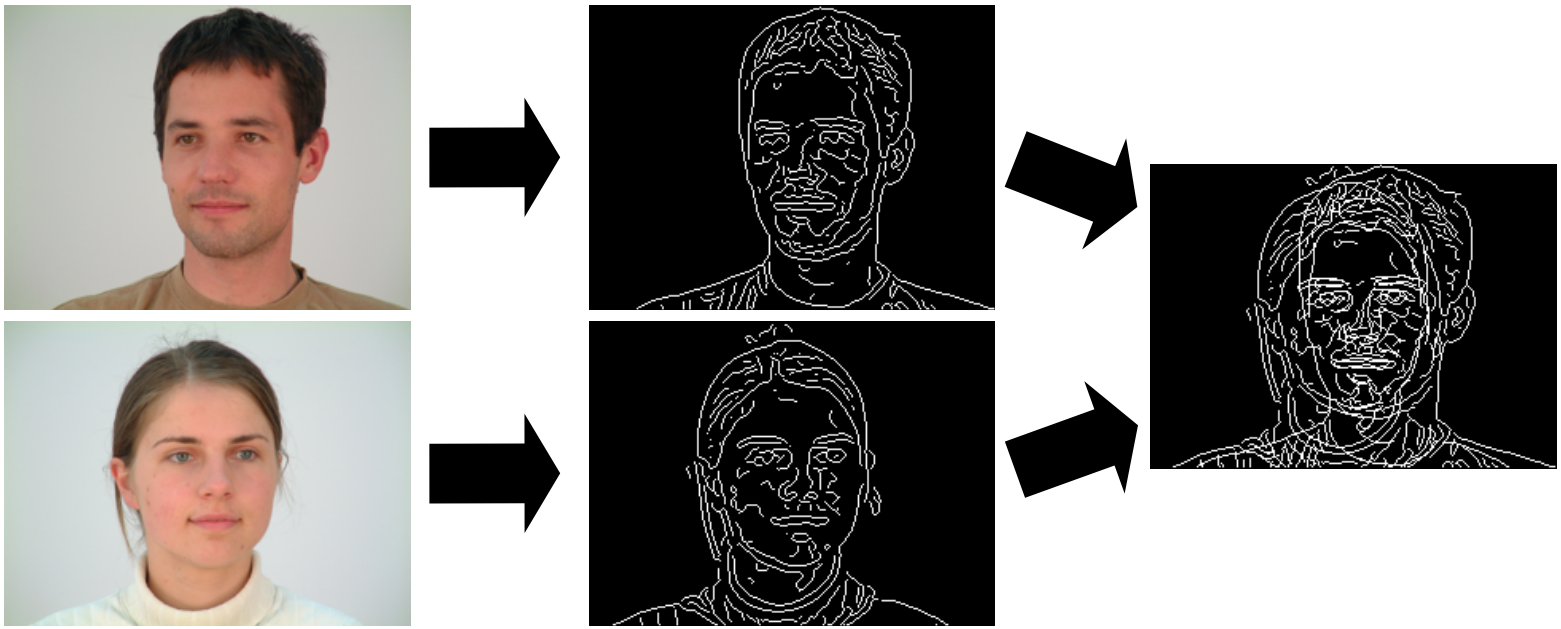
- Generation of Hough spaces for training images, using initial GHT model
- Separation of Hough space votes coming from every single model point: $f_j(c_i, X_n)$
- Weighted recombination of model point contributions into Maximum Entropy Distribution : $p_\Lambda(c_i|X_n)$
- Optimization of introduced weights λ_j by using a Minimum Classification Error (MCE)

$$E(\Lambda) = \sum_{n=1}^{N} \sum_{i=1}^{I} \varepsilon(c_i, \tilde{c}_n) \cdot \frac{p_\Lambda(c_i|X_n)^\eta}{\sum_k p_\Lambda(c_k|X_n)^\eta}.$$

$\varepsilon(c_i, \tilde{c}_n)$:	Error measure (e.g. Euclidean Distance)
$\tilde{c}_n$:	Target cell in Hough space for image X_n
I:	Number of Hough cells
N:	Number of training images

Iterative training

- Initial Model generation
- Discriminative weighting of individual model points
- Model point selection based on absolute weights
- Eye detection on training images using weighted model
- Select images with high localization error & incorporate image structures into model



Iterative training

- Initial Model generation
- Discriminative weighting of individual model points
- Model point selection based on absolute weights
- Eye detection on training images using weighted model
- Select images with high localization error & incorporate image structures into model

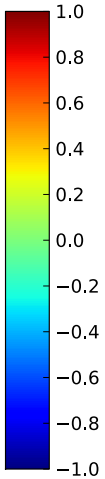


Initial GHT model



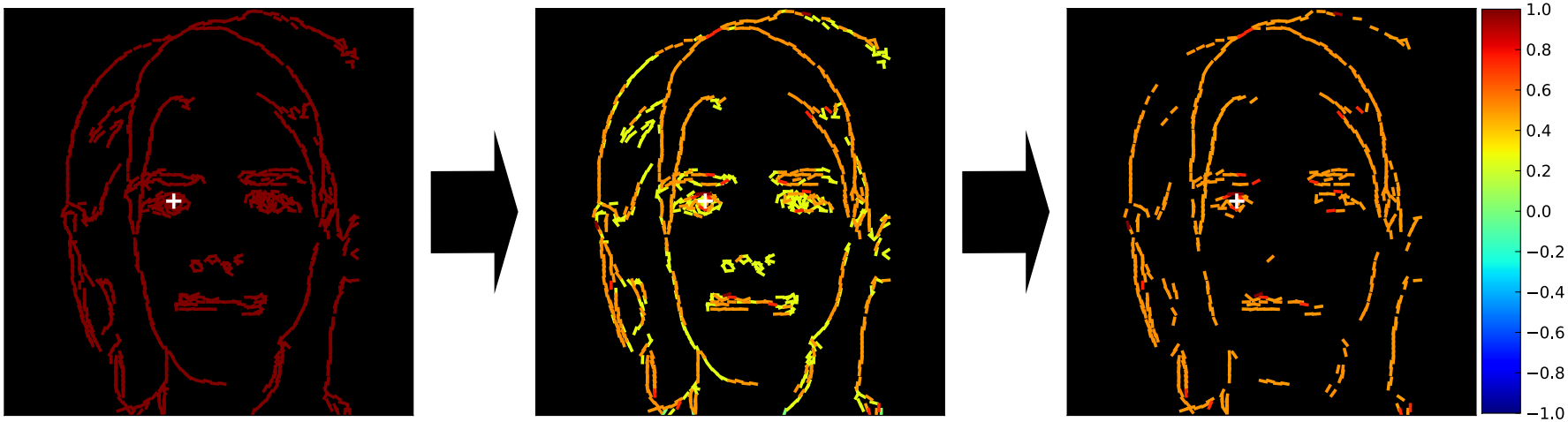
Weighted GHT model

$$p_{\Lambda}(c_i|X_n) = \frac{\exp\left(\sum_j \lambda_j \cdot f_j(c_i, X_n)\right)}{\sum_k \exp\left(\sum_j \lambda_j \cdot f_j(c_k, X_n)\right)}$$



Iterative training

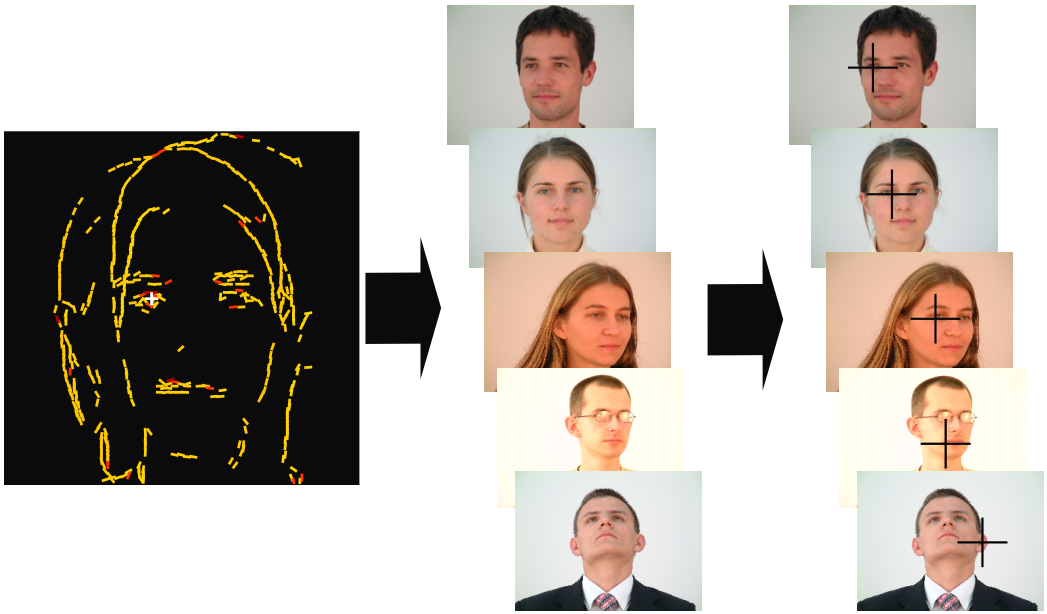
- Initial Model generation
- Discriminative weighting of individual model points
- Model point selection based on absolute weights
- Eye detection on training images using weighted model
- Select images with high localization error & incorporate image structures into model



Initial GHT model | Weighted GHT model | Reduced GHT model

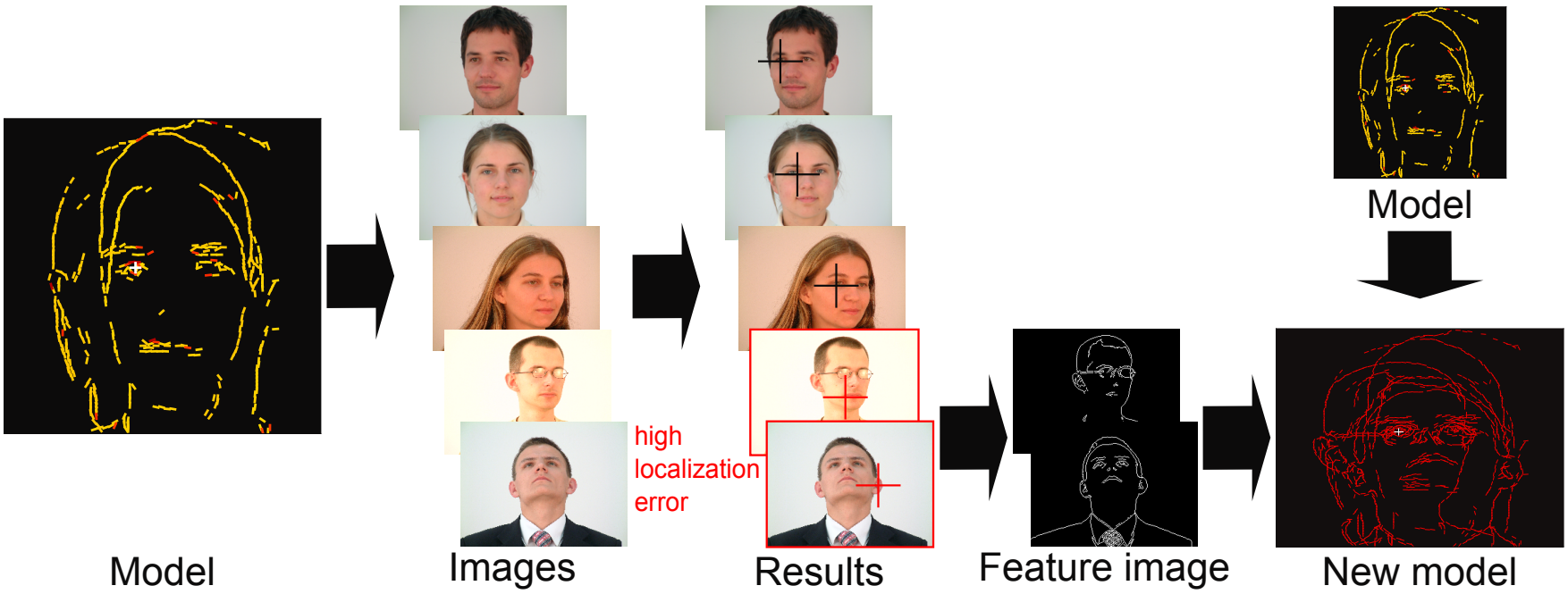
Iterative training

- Initial Model generation
- Discriminative weighting of individual model points
- Model point selection based on absolute weights
- Eye detection on training images using weighted model
- Select images with high localization error & incorporate image structures into model



Iterative training

- Initial Model generation
- Discriminative weighting of individual model points
- Model point selection based on absolute weights
- Eye detection on training images using weighted model
- Select images with high localization error & incorporate image structures into model



Multi-Level-Approach

- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point



Original image

Multi-Level-Approach

- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point



First Level: Reduced resolution

Multi-Level-Approach

- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point



Rough localization

Multi-Level-Approach

- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point

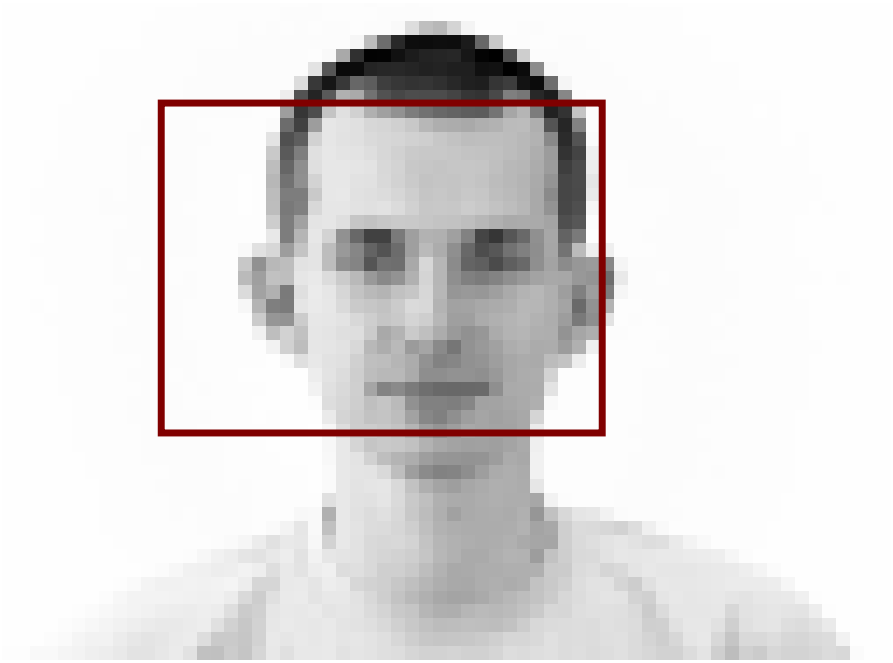
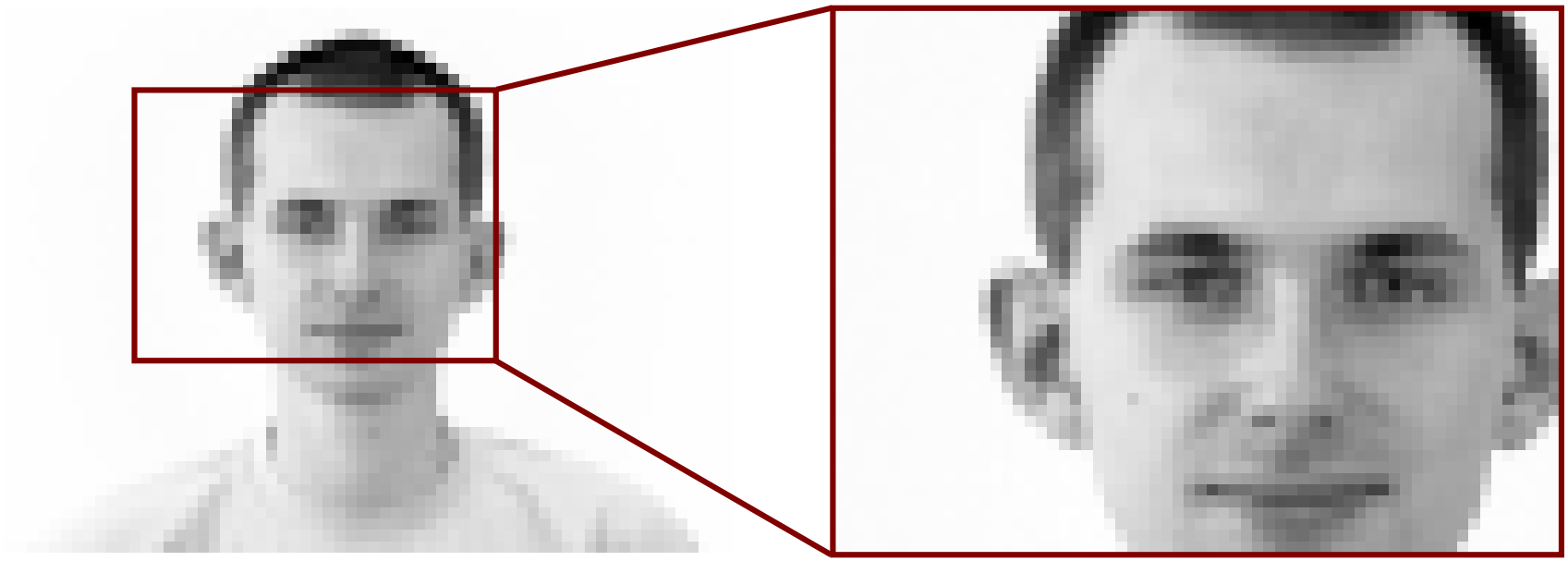


Image extract around detected point

Multi-Level-Approach

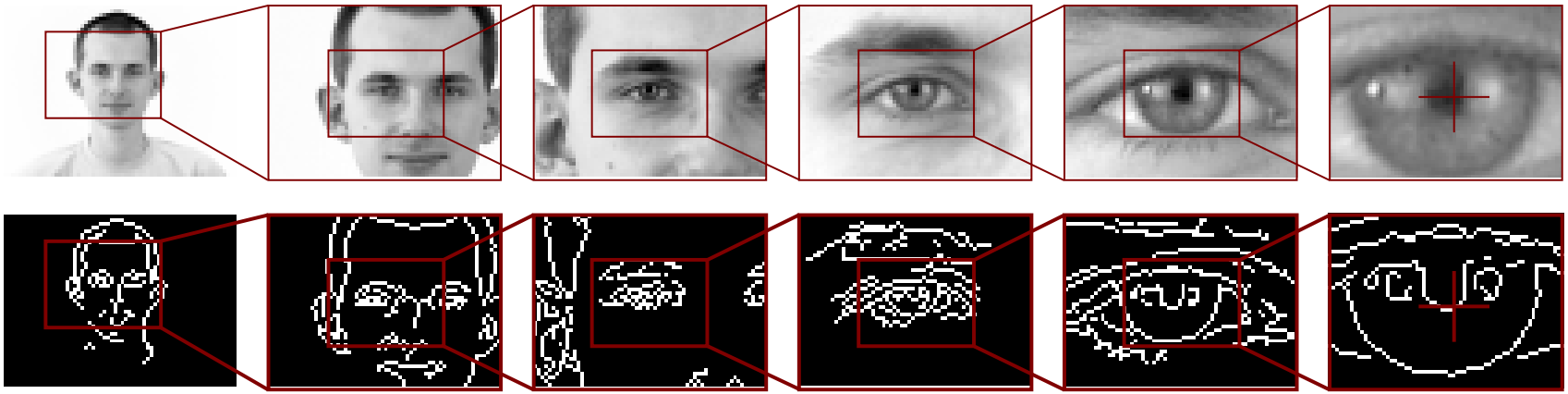
- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point



Next level: Higher resolution

Multi-Level-Approach

- Zooming-in strategy based on Gaussian pyramid
- Stepwise increase image resolution around detected point



- Specifically trained DGHT model on each level
  $\rightarrow$ Capturing of different level-specific details
- Good trade-off between model accuracy and reduced confusion with concurrent objects

Outline

1 Introduction

2 Method

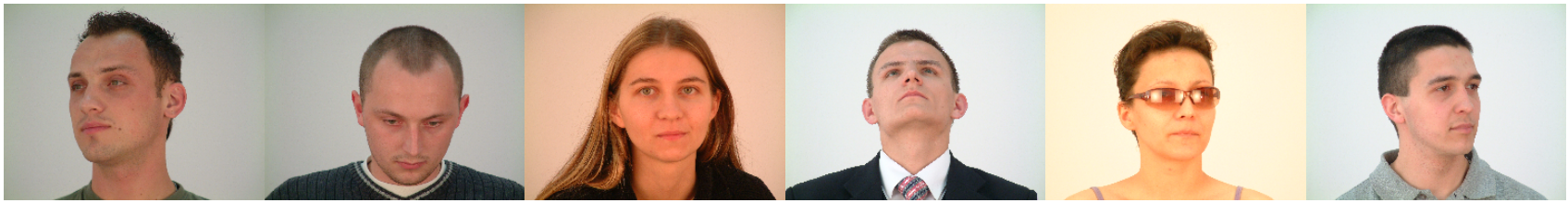
3 Experiments

4 Conclusion

Experimental Setup

Data:

- PUT Face Database [Kas08]
- 9971 face images of 100 subjects
- Several annotated face landmarks
- High resolution: 2048 x 1536
- Large variations in head pose
- Additional dataset with 22 frontal view images per person



[Kas08]: A. Kasinski et al., The PUT face database, Image Processing and Communications 13(3-4), 2008

Experimental Setup

- Random separation into training and evaluation subjects
- Training corpus: 60 subjects with 10 images per subject
  $\rightarrow$ 600 training images
- Evaluation on remaining 40 subjects
- Evaluation corpus: 3830 images
- Frontal view evaluation corpus: 869 images

Experimental Setup

- Feature generation: Canny edge detector [Can86]
- Multi-Level approach with 6 zoom levels
- Validation of localization success [Jes01]:
  Worst result for both eyes, normalized with eye distance
- Error $e < 0.1$: both localization results inside the irises
- Error $e < 0.25$: both localization results inside the eyes

[Can86]: J. Canny, A computational approach to edge detection, Pattern Analysis and Machine Intelligence 8(6), 1986
[Jes01]: O. Jesorsky et al., Robust face detection using the hausdorff distance, AVBPA Conference, 2001

Results

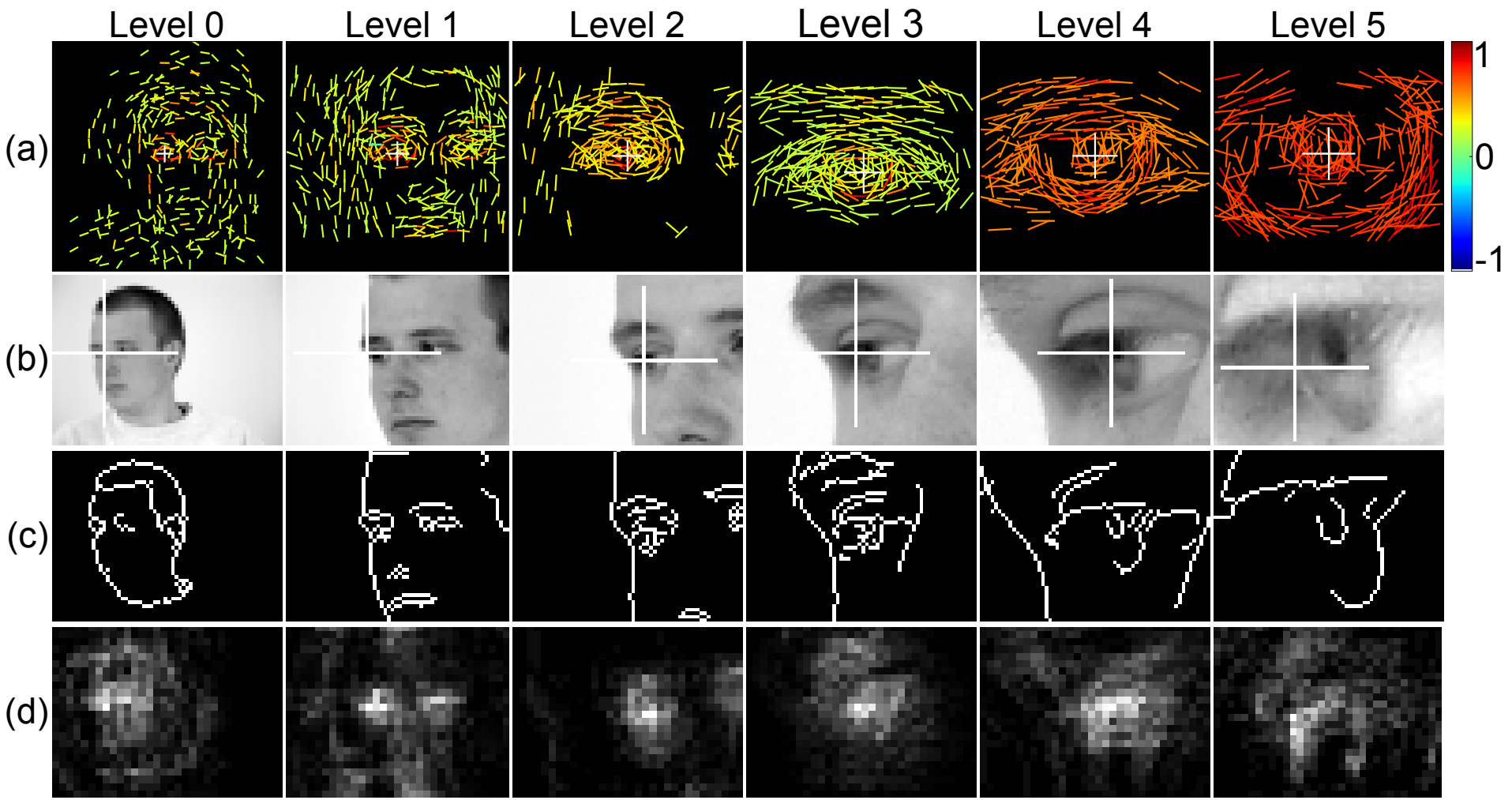
- Localization rate of 95% within the iris achieved.
- Current best result on this dataset: 94% (by [Kas10]).
- Average processing time for a single image: 597 ms

	$e < 0.1$	$e < 0.15$	$e < 0.2$	$e < 0.25$
frontal view subset				
DGHT	99.1%	99.1%	99.3%	99.4%
[Kas10]	99%			
full head pose variations				
DGHT	95.0%	95.4%	96.0%	96.5%
[Kas10]	94%			

[Kas10]: A. Kasinski et al., The architecture and performance of the face and eyes detection system based on the haar cascade classifiers, Pattern Analysis & Applications 13(2), 2010

Discussion

Example:



Conclusion

- Successful transfer of DGHT methodology to problem of eye localization
- Comparable results to state-of-the-art methods achieved
- General machine-learning approach, quickly adaptable to new tasks
- Minimal user interaction required during training
- Visual interpretation of generated models possible

Thank you!

Annotations provided by the PUT face database:

