

# Rozpoznávání tváří II

**Vojtěch Franc**

Centrum strojového vnímání, ČVUT FEL Praha

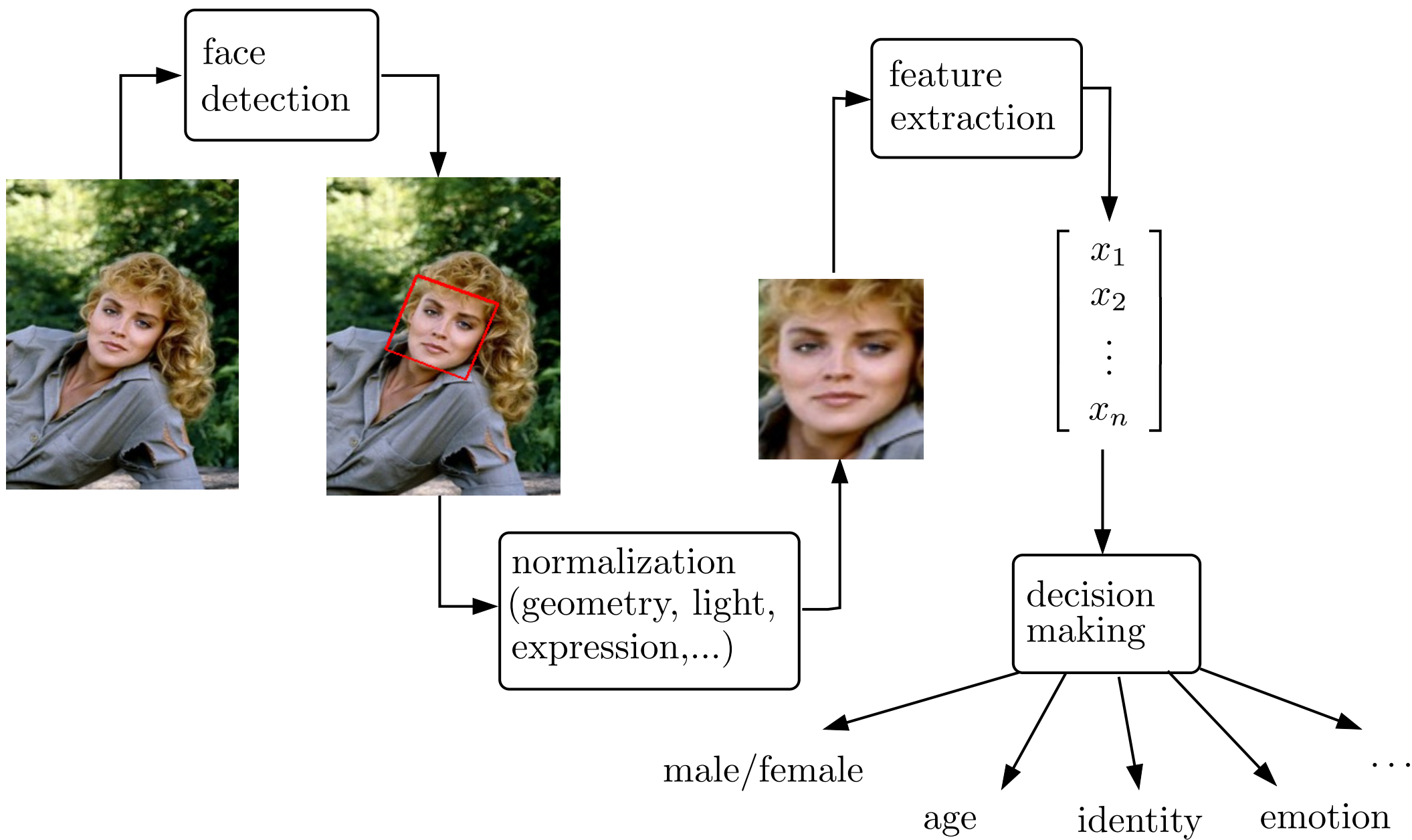


Biometrie ZS 2018

## **Osnova:**

- ◆ Registrace tváří.
- ◆ Repräsentace tváří.

# Stavební bloky typického systému pro rozpoznávání tváří

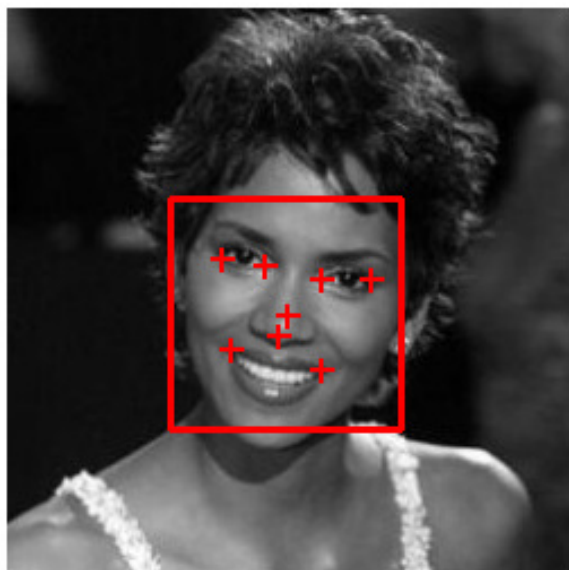


## Příklad geometrické normalizace tváře - affinní vyrovnání

- ◆ Cíl: Převést vstupní tvář do polohy, která bude nezávislá na třídě geometrických transformací, vůči které chceme být invariantní (např. rotace, translace, změna měřítka).
- ◆ Metoda zajišťující invarianci vůči affinní transformaci: nalezneme parametry affinní transformace (např. metodou nejmenších čtverců)

$$\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{b} \quad \mathbf{A} \in \mathbb{R}^{2 \times 2}, \mathbf{b} \in \mathbb{R}^{2 \times 1}, \mathbf{x} \in \mathbb{R}^{2 \times 1}$$

která promítne nalezené významné body  $(\mathbf{x}_1, \dots, \mathbf{x}_L)$  na jejich kanonickou konfiguraci  $(\mathbf{x}'_1, \dots, \mathbf{x}'_L)$ .

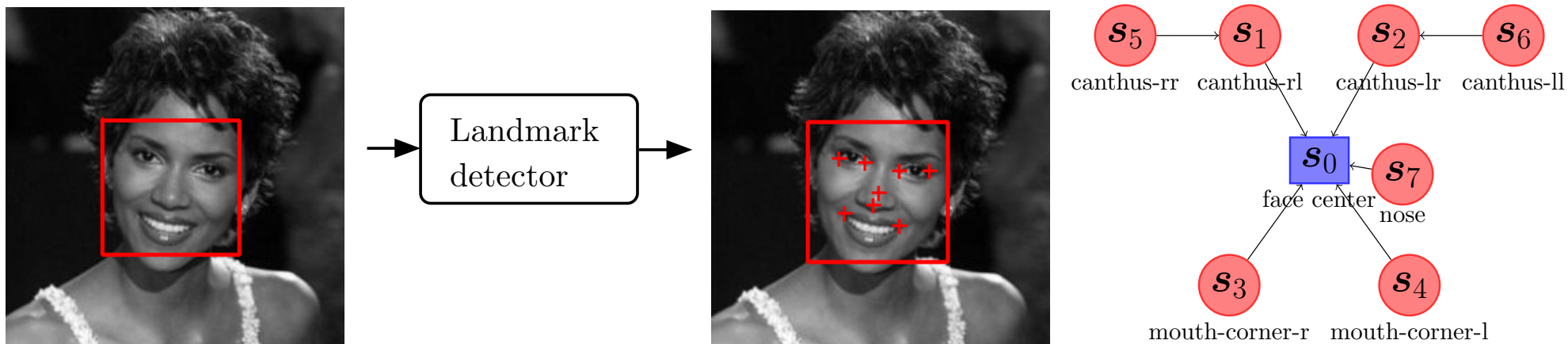


Vstupní obrázek



Normalizovaný obrázek

# Registrace tváře - Deformable Part Model detector významných bodů



**Strukturní klasifikátor** (deformable part model) odhaduje pozici významných bodů z obrázku:

$$(\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_L) = \underset{(\mathbf{x}_1, \dots, \mathbf{x}_L) \in \mathcal{N}^{2 \times L}}{\text{Argmax}} \text{skóre}(I, \mathbf{x}_1, \dots, \mathbf{x}_L)$$

kde skóre je dekomponované na součet jednoduchých funkcí

$$\text{skóre}(I, \mathbf{x}_1, \dots, \mathbf{x}_L) = \underbrace{\sum_{i \in V} f_i(I, \mathbf{x}_i)}_{\text{shoda s obrázkem}} + \underbrace{\sum_{ij \in E} f_{ij}(\mathbf{x}_i, \mathbf{x}_j)}_{\text{přípustnost tvaru}}$$

**Problém:** Jak při zafixované struktuře  $(V, E)$  zvolit funkce  $(f_i, i \in V)$  a  $(f_{ij}, ij \in E)$  tak, aby detektor fungoval dobře?

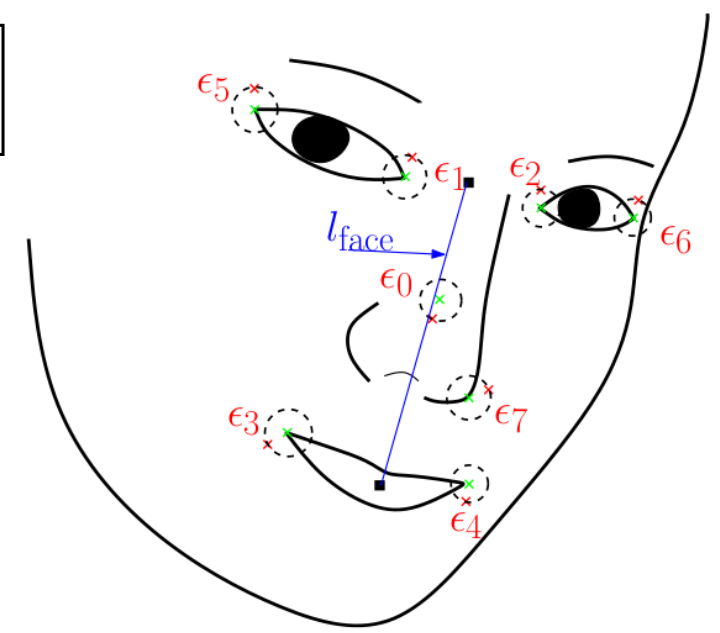
# Učení DPM detektoru

**Problém učení:** Nalezni  $f = \{(f_i, i \in V), (f_{ij}, ij \in E)\}$  tak, aby strukturní klasifikátor

$$(\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_L) = \underset{(\mathbf{x}_1, \dots, \mathbf{x}_L) \in \mathcal{N}^{2 \times L}}{\text{Argmax}} \left[ \sum_{i \in V} f_i(I, \mathbf{x}_i) + \sum_{ij \in E} f_{ij}(\mathbf{x}_i, \mathbf{x}_j) \right]$$

odhadoval co nejpřesněji, ve smyslu průměrné odchylky,

$$\text{err} = \frac{\epsilon_0 + \dots + \epsilon_{L-1}}{L} \cdot \frac{1}{l_{\text{face}}}$$



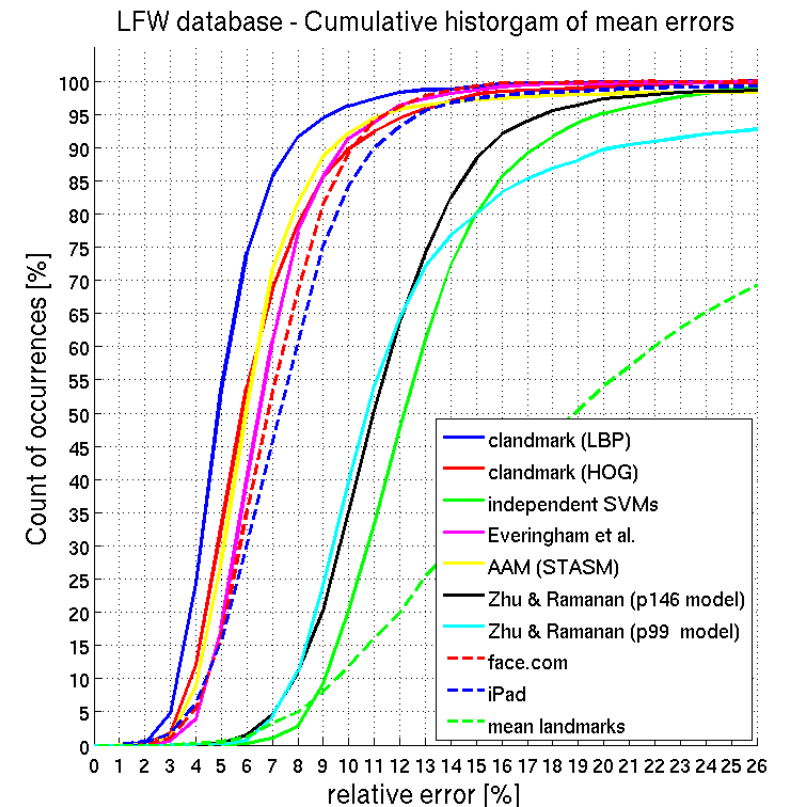
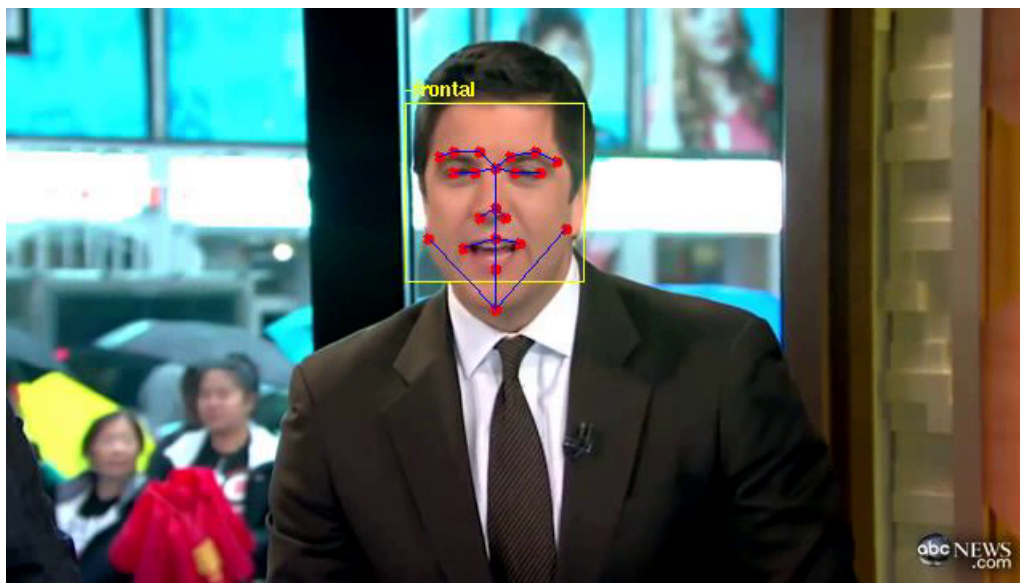
měření na manuálně označené trénovací sadě příkladů:

$$E(f) = \text{err}_1(f) + \text{err}_2(f) + \text{err}_3(f) + \text{err}_4(f) + \text{err}_5(f) + \dots + \text{err}_m(f)$$



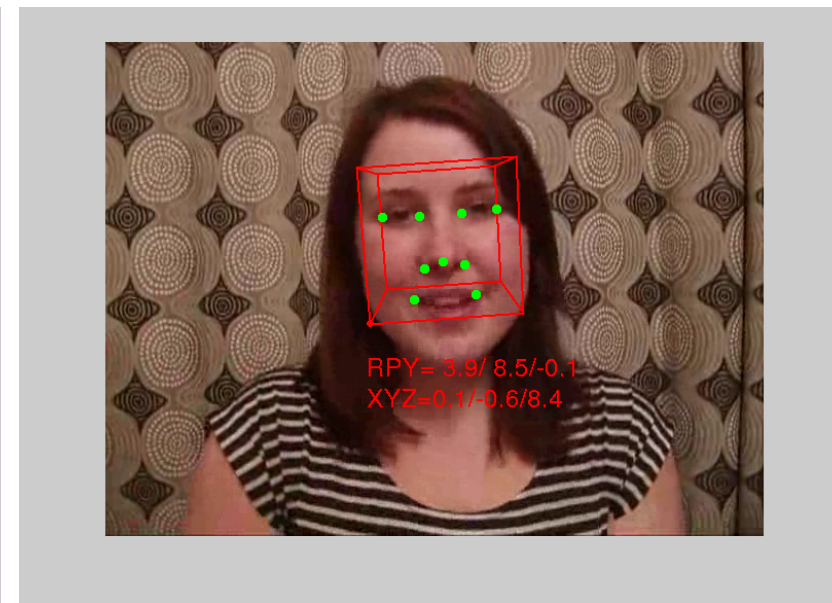
**Structured output SVM:** Úlohu učení  $\min_f E(f)$  převede na problém minimalizace konvexní funkce, kterou lze efektivně řešit i pro velká data.

- ◆ Language: C++, binding to Matlab
- ◆ Automatizovaný návrh: popiš strukturu v XML, sesbírej příklady a spust' učení
- ◆ Multi-platform: PC, iPad, Geod, ARM
- ◆ Real-time:  $\approx 3\text{ms}$  per face on ordinary PC



- ◆ Download: <http://cmp.felk.cvut.cz/~uricamic/clandmark/>

# Registrace tváře - 3D detektor významných bodů



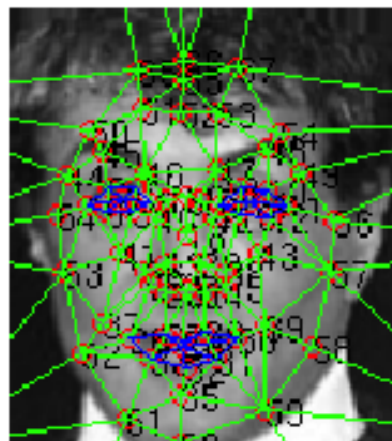
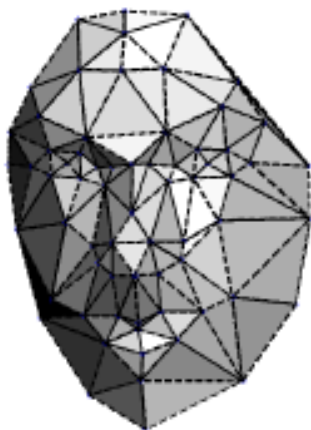
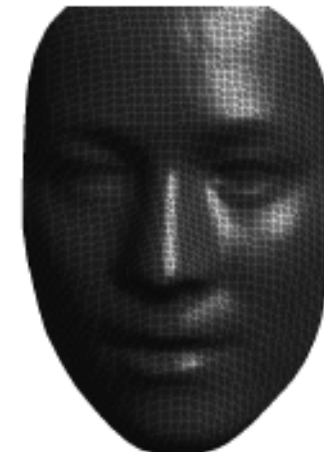
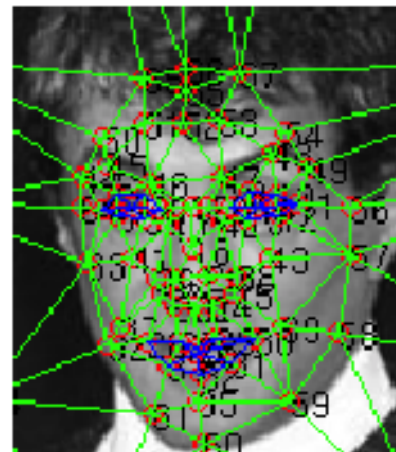
Fitování 3D modelu tváře do 2D obrázku:

$$(\hat{\mathbf{t}}, \hat{\alpha}, \hat{\beta}, \hat{\gamma}) = \underset{(\mathbf{t}, \alpha, \beta, \gamma) \in \mathbb{R}^6}{\text{Argmax}} \text{score}(I, \mathbf{t}, \alpha, \beta, \gamma) := \sum_{i=1}^L f_i \left( I, K[R(\alpha, \beta, \gamma) | \mathbf{t}] \mathbf{X}_i \right)$$

$(\mathbf{X}_1, \dots, \mathbf{X}_L) \in \mathbb{R}^{3 \times L}$     3D model tváře  
 $K[R(\alpha, \beta, \gamma) | \mathbf{t}]$     model perspektivní kamery  
 $(\mathbf{t}, \alpha, \beta, \gamma)$     poloha a úhly natočení hlavy

## Příklad geometrické normalizace tváře - frontalizace

- ◆ 3D model se použije k transformaci významných bodů detekovaných ve vstupním obrázku do konfigurace, kterou bychom viděli při frontálním pohledu na tvář.





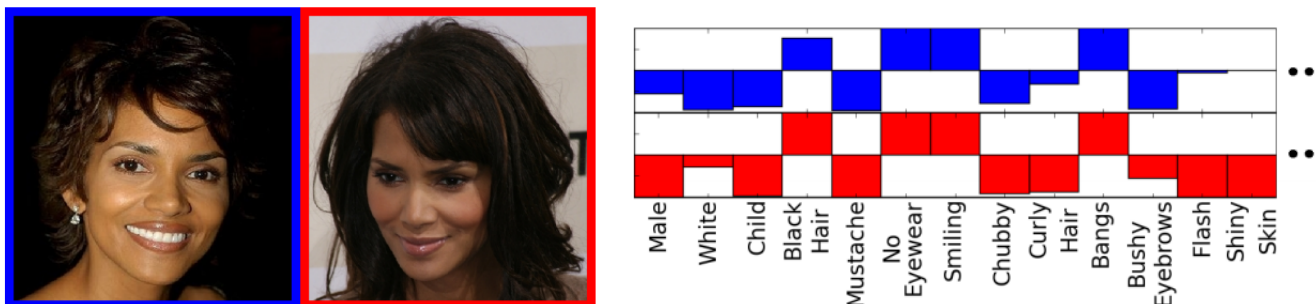
# Numerická reprezentace tváře

**Era I:** obrázek je reprezentován jako vektor v lineárním prostoru - PCA, LDA, ...

$$\mathbf{x} \quad \tilde{\mathbf{x}} = \mu + z_1 \mathbf{a}_1 + z_2 \mathbf{a}_2 + z_3 \mathbf{a}_3 + z_4 \mathbf{a}_4 + z_5 \mathbf{a}_5 + z_6 \mathbf{a}_6 + z_7 \mathbf{a}_7 + z_8 \mathbf{a}_8$$

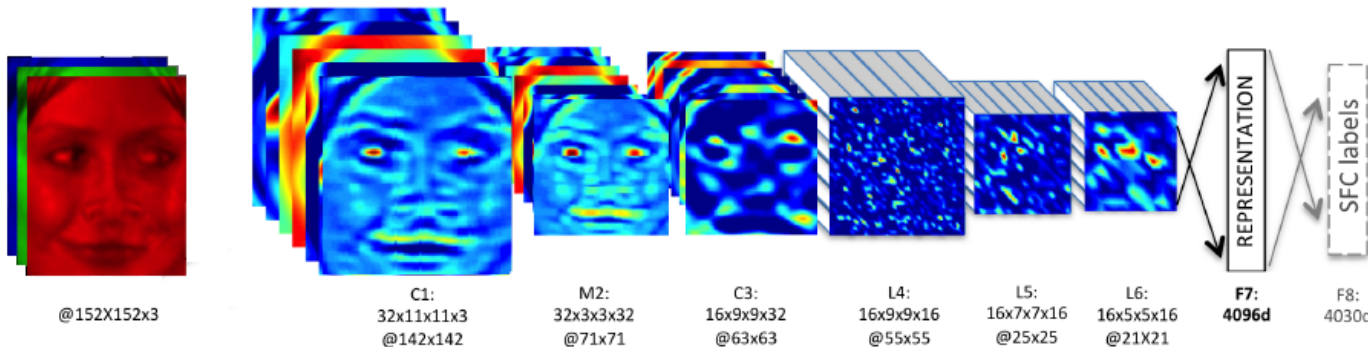
*Turk & Pentland: Eigenfaces for Recognition. Jour. Cogn. Neuro. 1991.*

**Era II:** Deskriptory navržené specificky pro popis tváře např. attribute descriptor



*Kumar et al: Attribute and Simile Classifiers for Face Verification. ICCV'09*

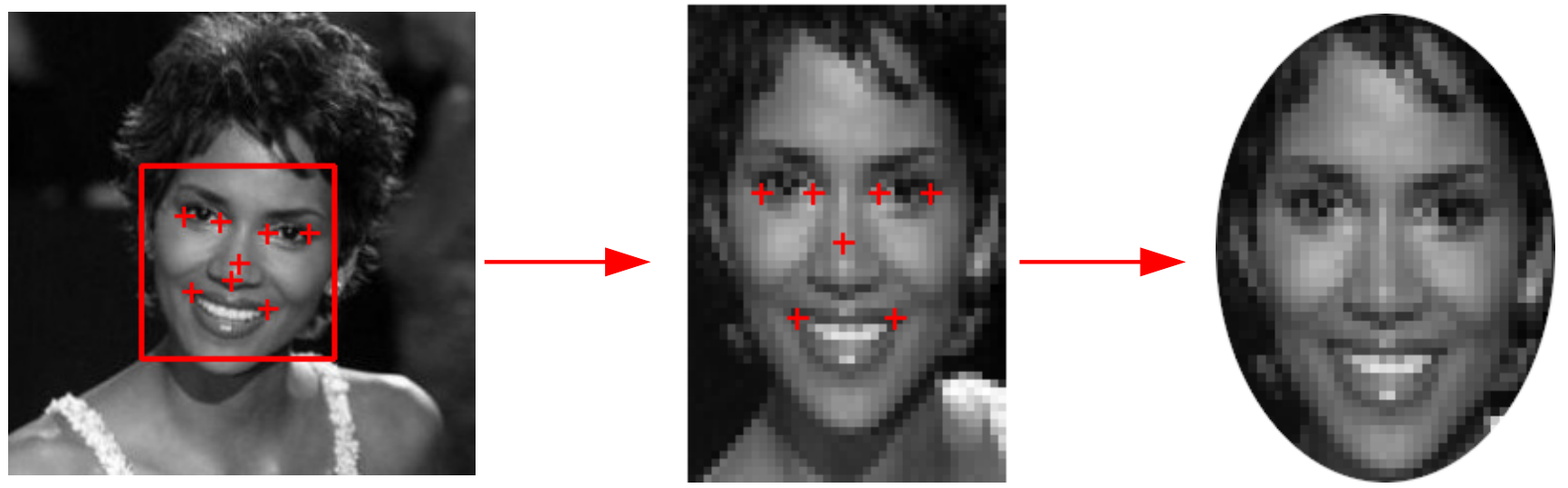
**Era III:** Reprezentace učená z dat např. pomocí hlubokých neuronových sítí



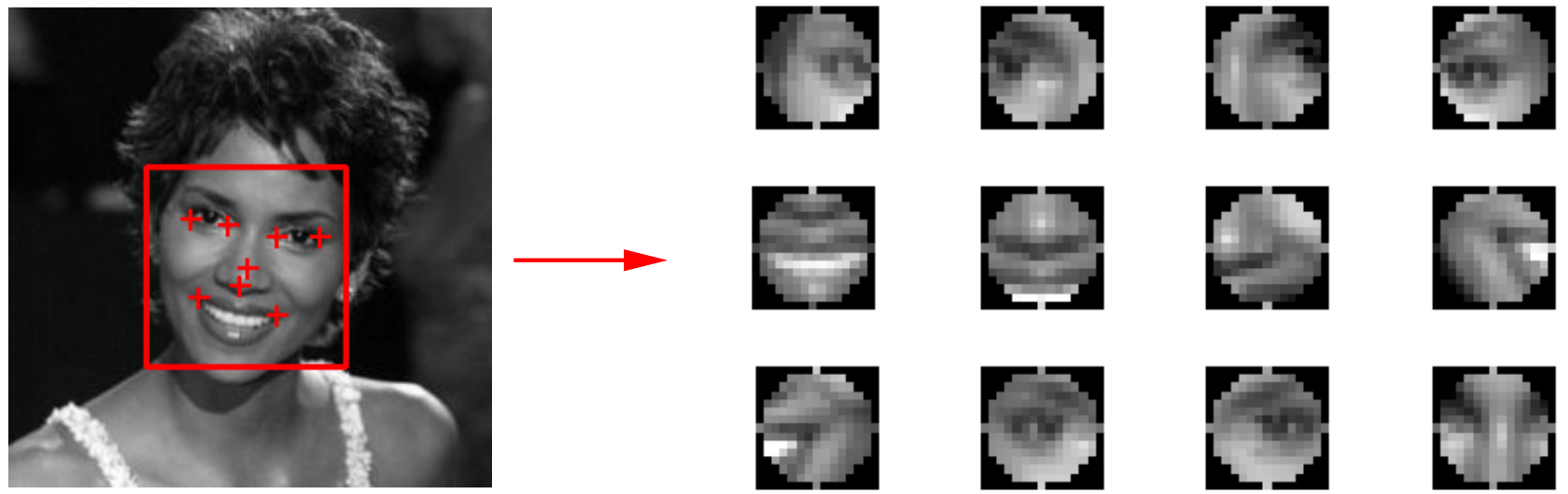
*Taigman et al: DeepFace: Closing the Gap to Human-Level Performance in Face Verification. CVPR 2014.*

# Holistický popis vs. lokální popis

- ◆ **Holistický popis:** celý výřez geometricky normalizované tváře.

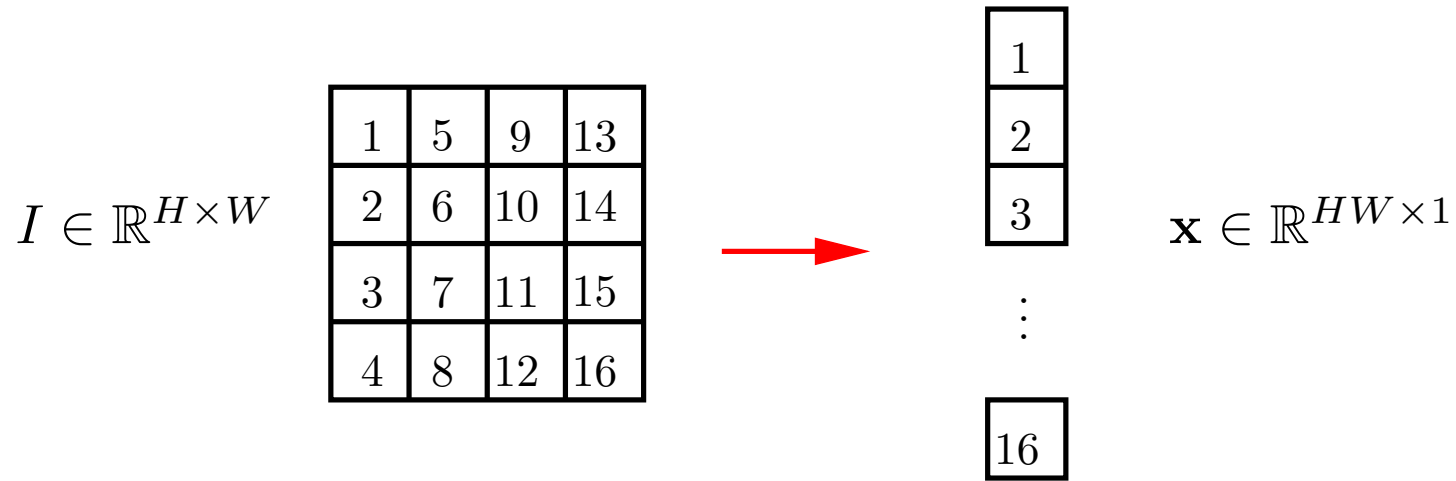


- ◆ **Lokální popis:** geometricky normalizované výřezy kolem významných bodů.



# Příznaková reprezentace obrázku pomocí jasových hodnot

- ◆ Nejjednodušší způsob reprezentace je použít přímo jasové hodnoty v jednotlivých pixelech jako příznaky



- ◆ Nevýhody: vysoká dimenze a citlivost na změnu jasů, pozice, výrazu ... všeho.
- ◆ Pro zvýšení invariance vůči změně osvětlení se používají jasové normalizace. Např:
  - Normalizace na nulovou střední hodnotu a jednotkovou varianci:

$$I'_{ij} = \frac{I_{ij} - \mu}{\sigma}, \quad \mu = \frac{1}{WH} \sum_{i=1}^W \sum_{j=1}^H I_{ij}, \quad \sigma = \frac{1}{WH} \sum_{i=1}^W \sum_{j=1}^H (I_{ij} - \mu)^2$$

- Ekvalizace histogramu.

# Principal Component Analysis: formulace

- ◆ Cíl: zadané body  $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \in \mathbb{R}^{n \times m}$  aproximovat v  $p$ -dimenzionálním affinním podprostoru, který je daný bází  $A = \{\mathbf{a}_1, \dots, \mathbf{a}_p\} \in \mathbb{R}^{n \times p}$  a středem  $\mu \in \mathbb{R}^n$ :

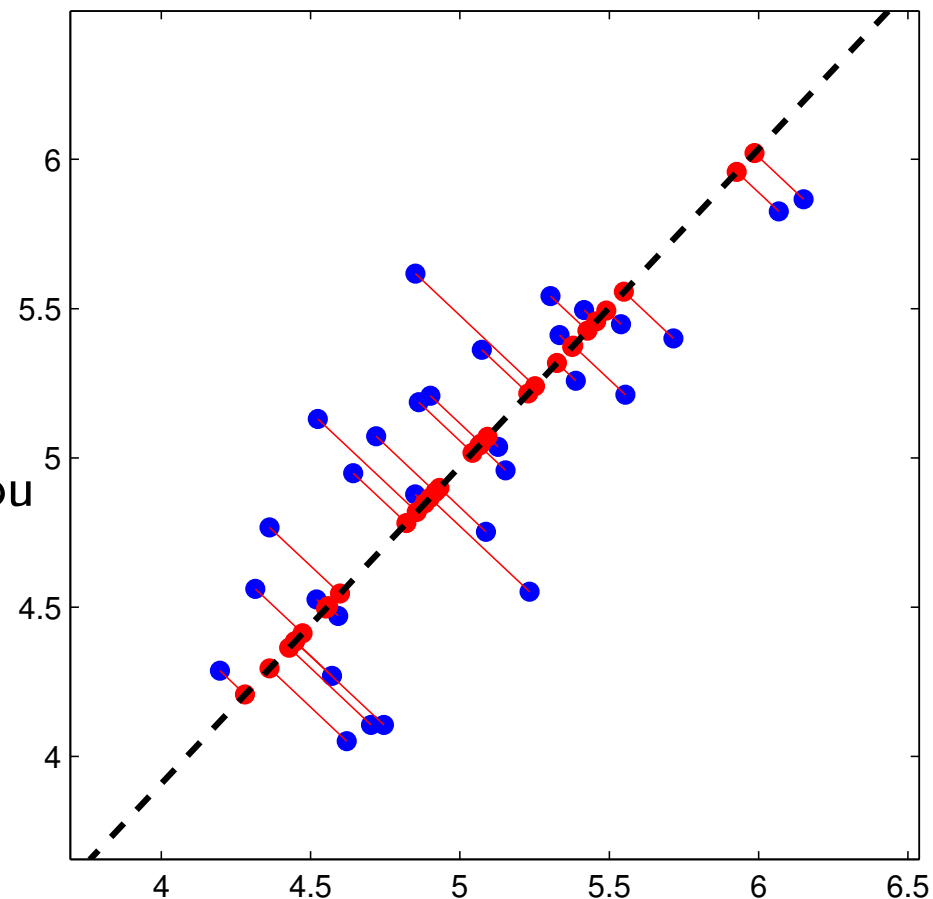
$$\tilde{X} = \left\{ \tilde{\mathbf{x}} \in \mathbb{R}^n \mid \tilde{\mathbf{x}} = \sum_{i=1}^p z_i \mathbf{a}_i + \mu \right\} \subseteq \mathbb{R}^n$$

- ◆ Projekce bodu  $\mathbf{x} \in \mathbb{R}^n$  na podprostor  $\tilde{X}$

$$\tilde{\mathbf{x}} = \operatorname{argmin}_{\mathbf{x}' \in \tilde{X}} \|\mathbf{x} - \mathbf{x}'\|$$

- ◆ Podprostor  $\tilde{X}$  s minimální aproximační chybou

$$E(\tilde{X}) = \sum_{i=1}^m \|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2$$



# Principal Component Analysis: řešení

- ◆ Řešením je affinní podprostor jehož bázové vektory  $A = \{\mathbf{a}_1, \dots, \mathbf{a}_p\}$  tvoří  $p$  vlastních vektorů s největším vlastním číslem spočtené z kovarianční matice

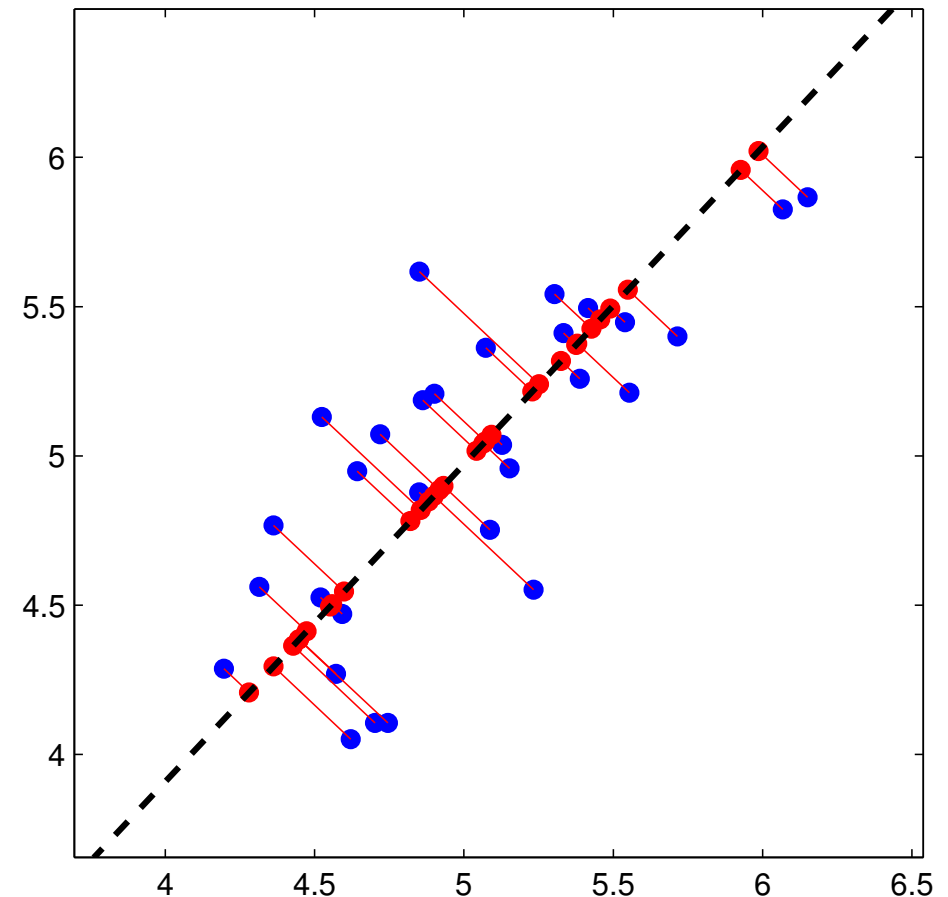
$$\mathbf{C} = \frac{1}{m} \sum_{i=1}^m (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T \quad \text{kde} \quad \mu = \frac{1}{m} \sum_{i=1}^m \mathbf{x}_i$$

- ◆ Projekce bodu  $\mathbf{x}$  na PCA prostor

$$\mathbf{z} = \mathbf{A}^T (\mathbf{x} - \mu)$$

- ◆ Zpětná projekce

$$\tilde{\mathbf{x}} = \mathbf{A}\mathbf{z} + \mu = \sum_{i=1}^p z_i \mathbf{a}_i + \mu$$

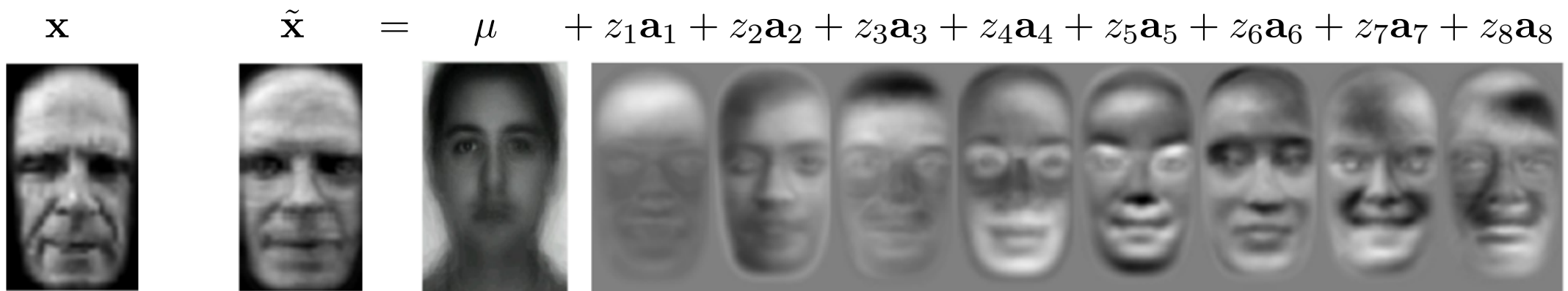


# Použití PCA pro reprezentaci tváří: Eigenfaces

- ◆ Tvář  $\mathbf{x}$  se promítne do  $p$ -dimenzionálního PCA prostoru

$$\mathbf{z} = \mathbf{A}^T(\mathbf{x} - \mu) = (\underbrace{\mathbf{a}_1^T(\mathbf{x} - \mu)}_{z_1}, \dots, \underbrace{\mathbf{a}_p^T(\mathbf{x} - \mu)}_{z_p})^T$$

- ◆ Tvář  $\mathbf{x}$  lze aproximovat jako lineární kombinaci  $p$  bázových vektorů (“eigen tváří”)



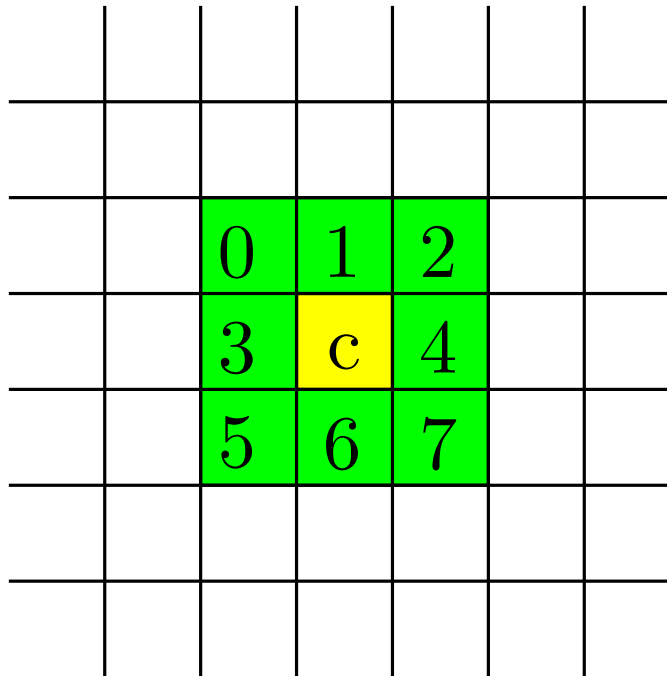
Každý obrázek  $\mathbf{x} \in \mathbb{R}^{H \times W}$  lze reprezentovat pomocí  $H \cdot W$  čísel.

Každý obrázek  $\tilde{\mathbf{x}} \in \mathbb{R}^{H \times W}$  lze reprezentovat pomocí  $p = 8$  čísel.

# Příklad reprezentace pomocí lokálních deskriptorů: Local Binary Patterns

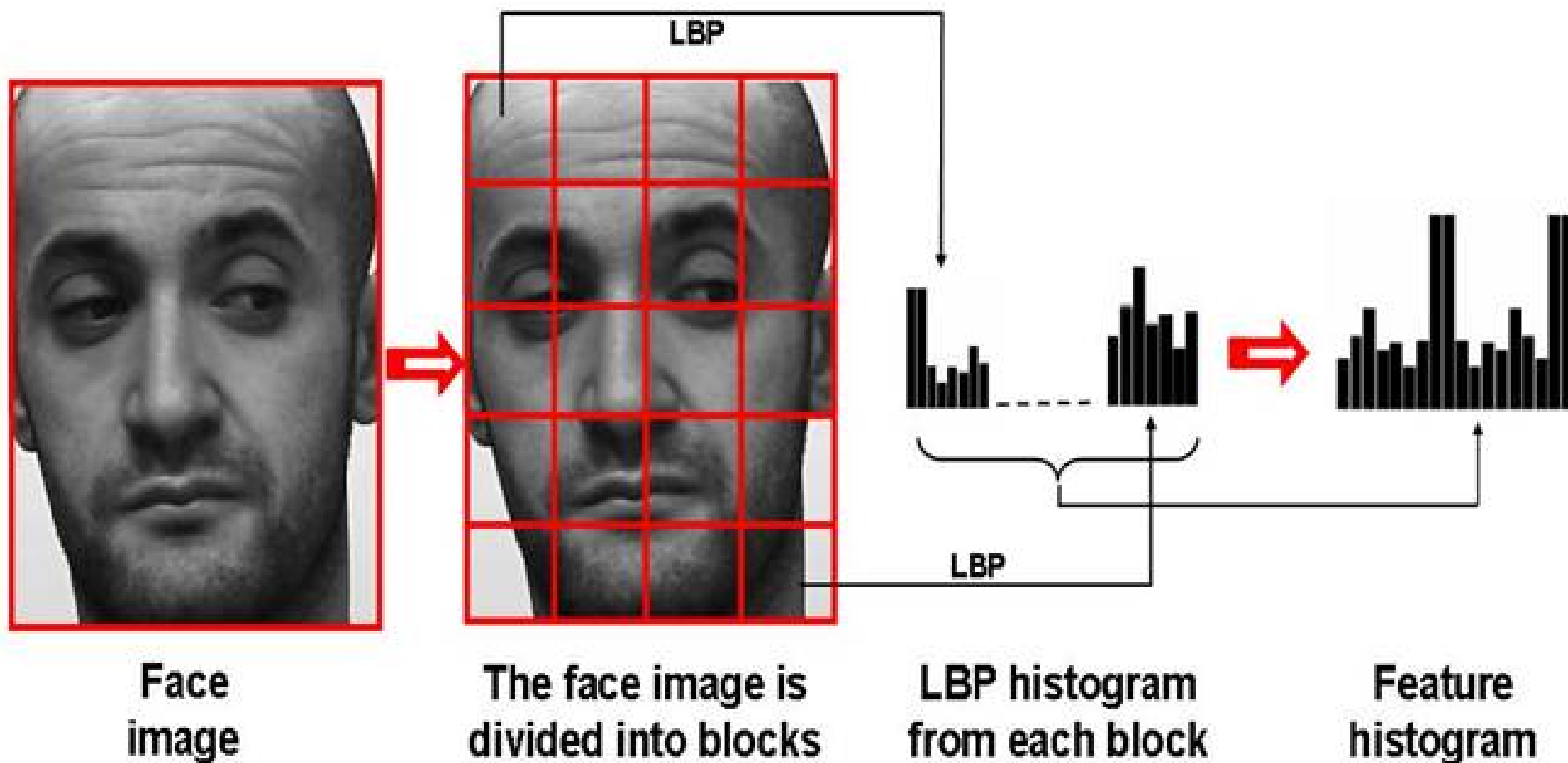
- ◆ LBP přiřadí oknu velikosti  $3 \times 3$  pixelů 8-bitový kód

$$LBP = \sum_{p=0}^7 [I(c) \geq I(p)] 2^p$$



- ◆ LBP příznaky jsou invariantní vůči monotónní změně osvětlení obrázku.

# Příklad reprezentace pomocí lokálních deskriptorů: Local Binary Patterns



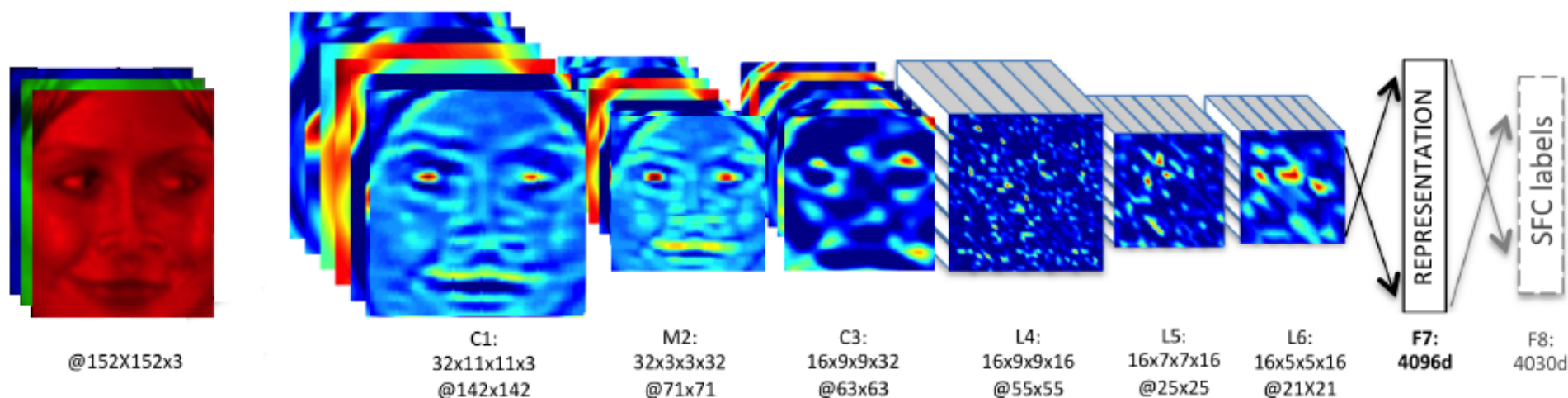


# Příznaková reprezentace učená z příkladů pomocí hlubokých konvolučních neuronových sítí



17/18

- ◆ **Konvoluční neuronová síť (CNN):** Vstupní obrázek se postupně transformuje pomocí operací konvoluce, ReLU a max-pooling. Poslední vrstvy tvoří Perceptronová síť.



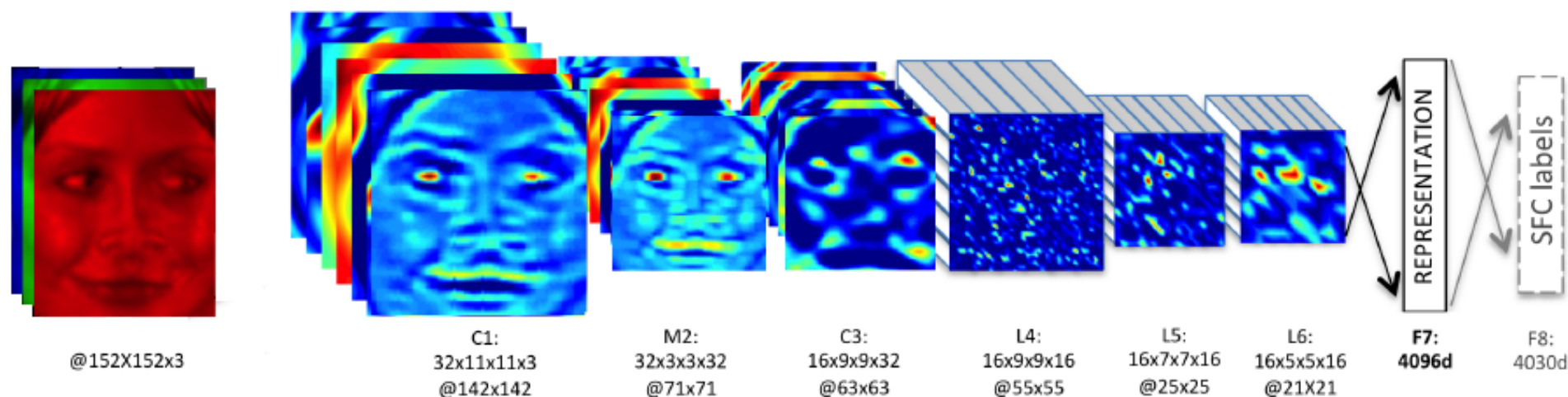
Obrázek převzat z Taigman et al: *DeepFace: Closing the Gap to Human-Level Performance in Face Verification*. CVPR 2014.

# Příznaková reprezentace učená z příkladů pomocí hlubokých konvolučních neuronových sítí



17/18

- ◆ **Konvoluční neuronová síť (CNN):** Vstupní obrázek se postupně transformuje pomocí operací konvoluce, ReLU a max-pooling. Poslední vrstvy tvoří Perceptronová síť.

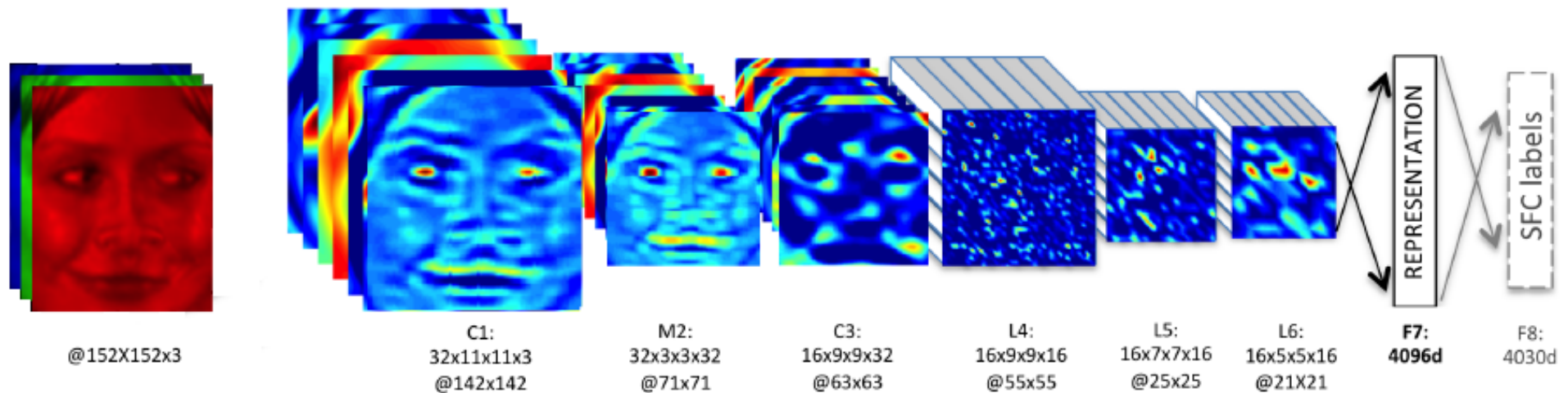


Obrázek převzat z Taigman et al: *DeepFace: Closing the Gap to Human-Level Performance in Face Verification*. CVPR 2014.

- ◆ **Učení s učitelem:** nauč síť tak, aby na dané sadě příkladů co nejlépe napodobovala učitele při řešení cílové úlohy nebo úlohy jí podobné.
  - Učení vede na nekonvexní optimalizační úlohu, která se řeší přibližně pomocí stochastické aproximace.
  - Pro učení je potřeba velké množství (milióny) anotovaných obrázků.

# Příznaková reprezentace učená z příkladů pomocí hlubokých konvolučních neuronových sítí

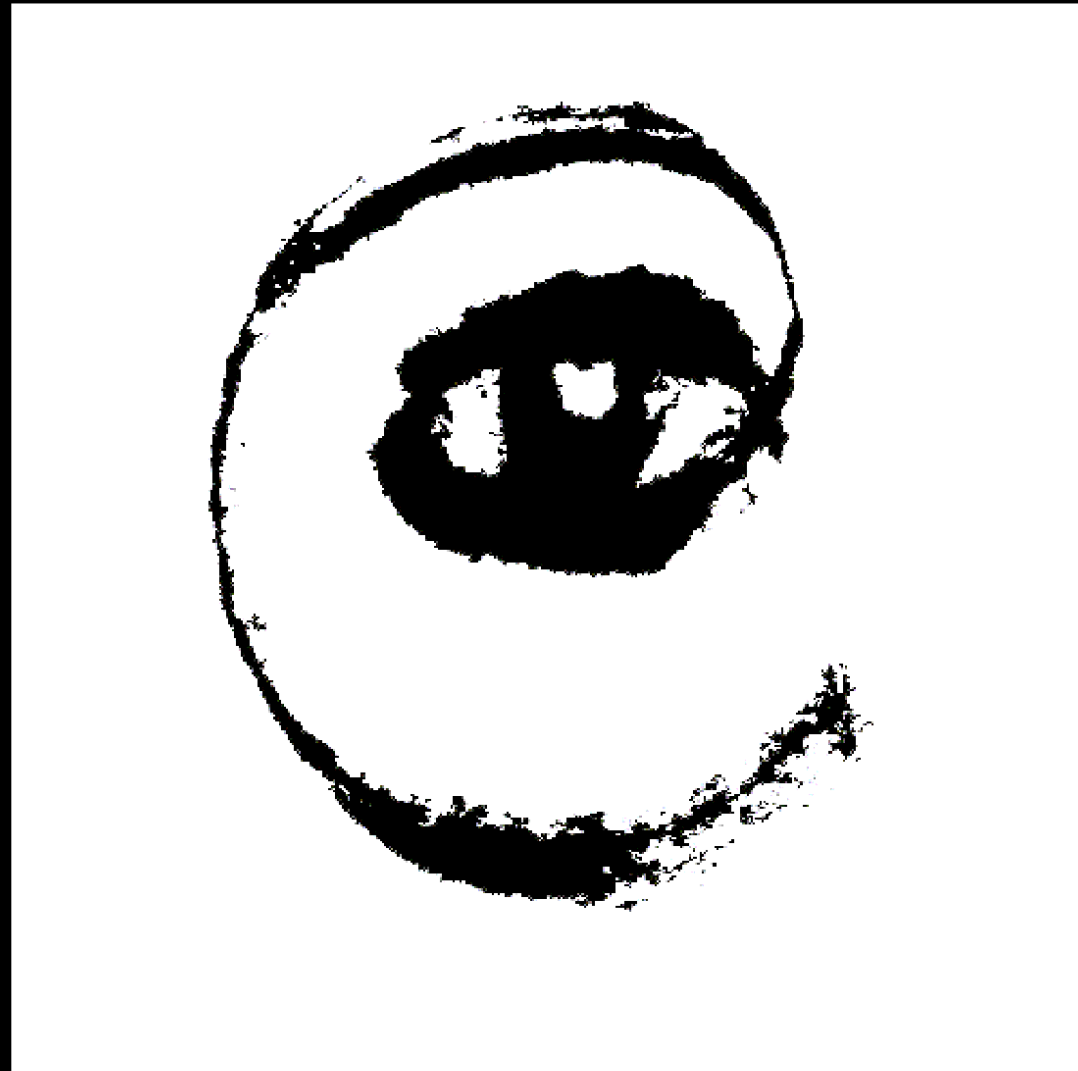
- ◆ **Konvoluční neuronová síť (CNN):** Vstupní obrázek se postupně transformuje pomocí operací konvoluce, ReLU a max-pooling. Poslední vrstvy tvoří Perceptronová síť.



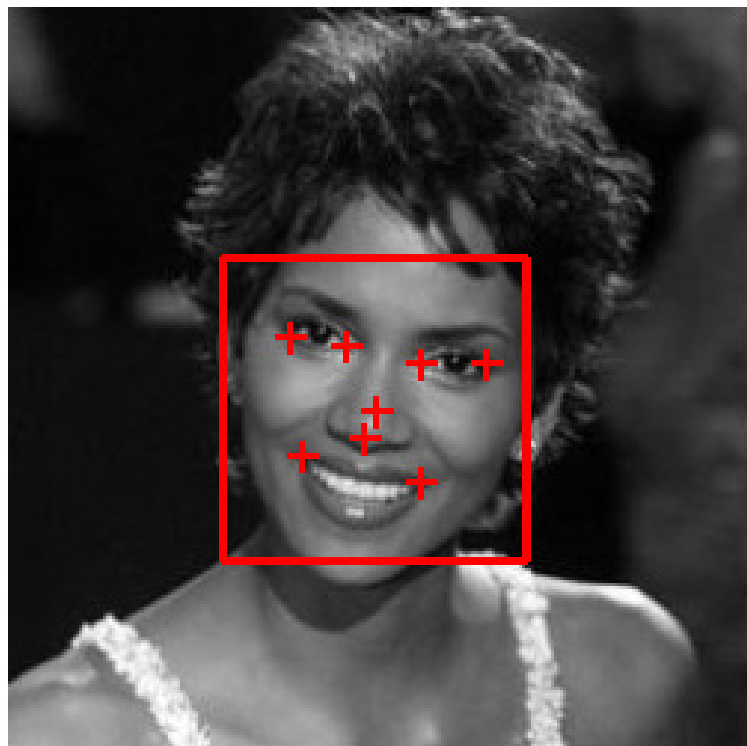
Obrázek převzat z *Taigman et al: DeepFace: Closing the Gap to Human-Level Performance in Face Verification. CVPR 2014.*

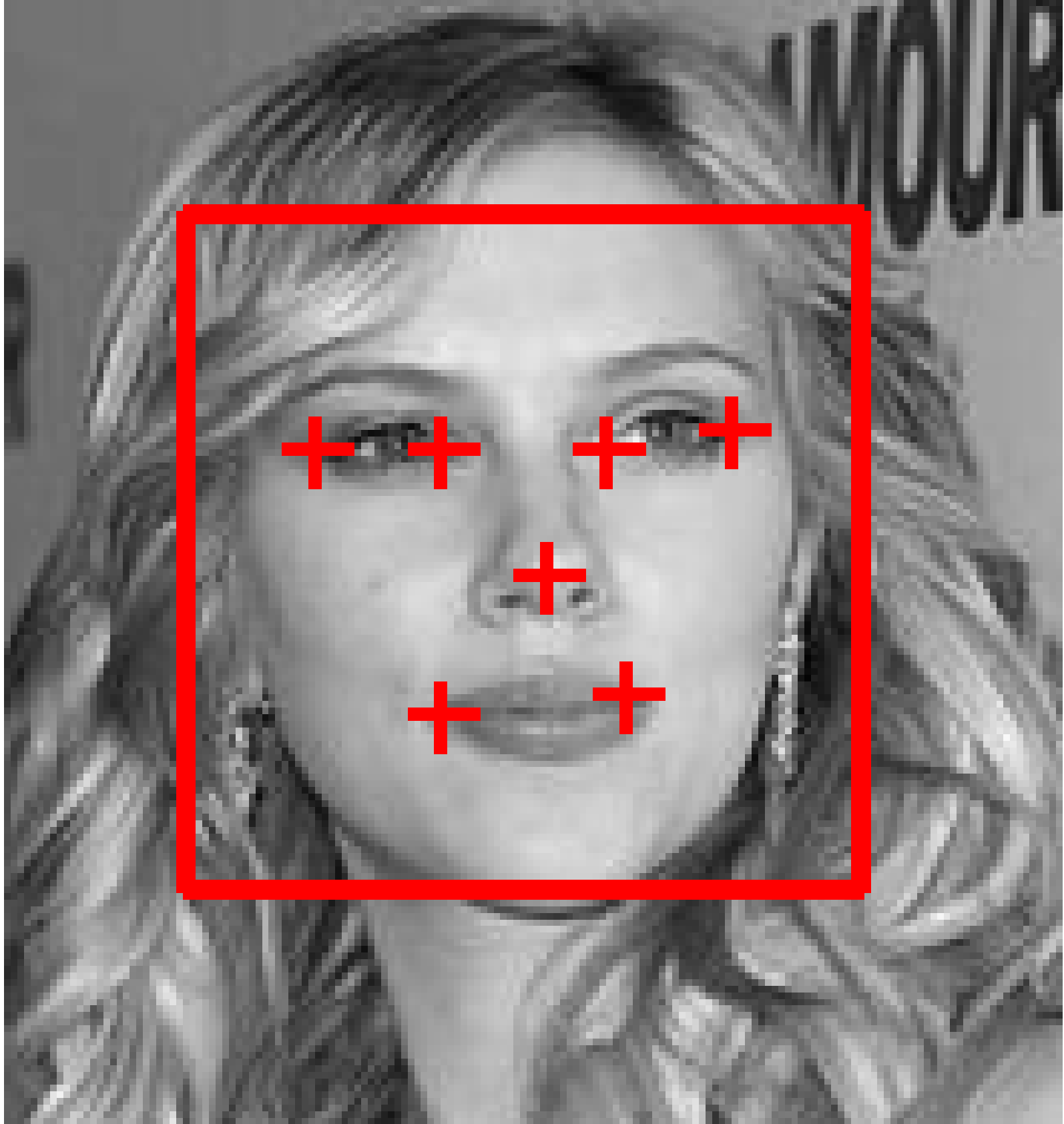
- ◆ **Učení s učitelem:** nauč síť tak, aby na dané sadě příkladů co nejlépe napodobovala učitele při řešení cílové úlohy nebo úlohy jí podobné.
  - Učení vede na nekonvexní optimalizační úlohu, která se řeší přibližně pomocí stochastické aproximace.
  - Pro učení je potřeba velké množství (milióny) anotovaných obrázků.
- ◆ **Transfer learning:** Řeší se podobná uloha, pro kterou je dostatek dat. Výstupy předposlední vrstvy se použijí jako příznakový popis reprezentující vstupní tvář.

Konec



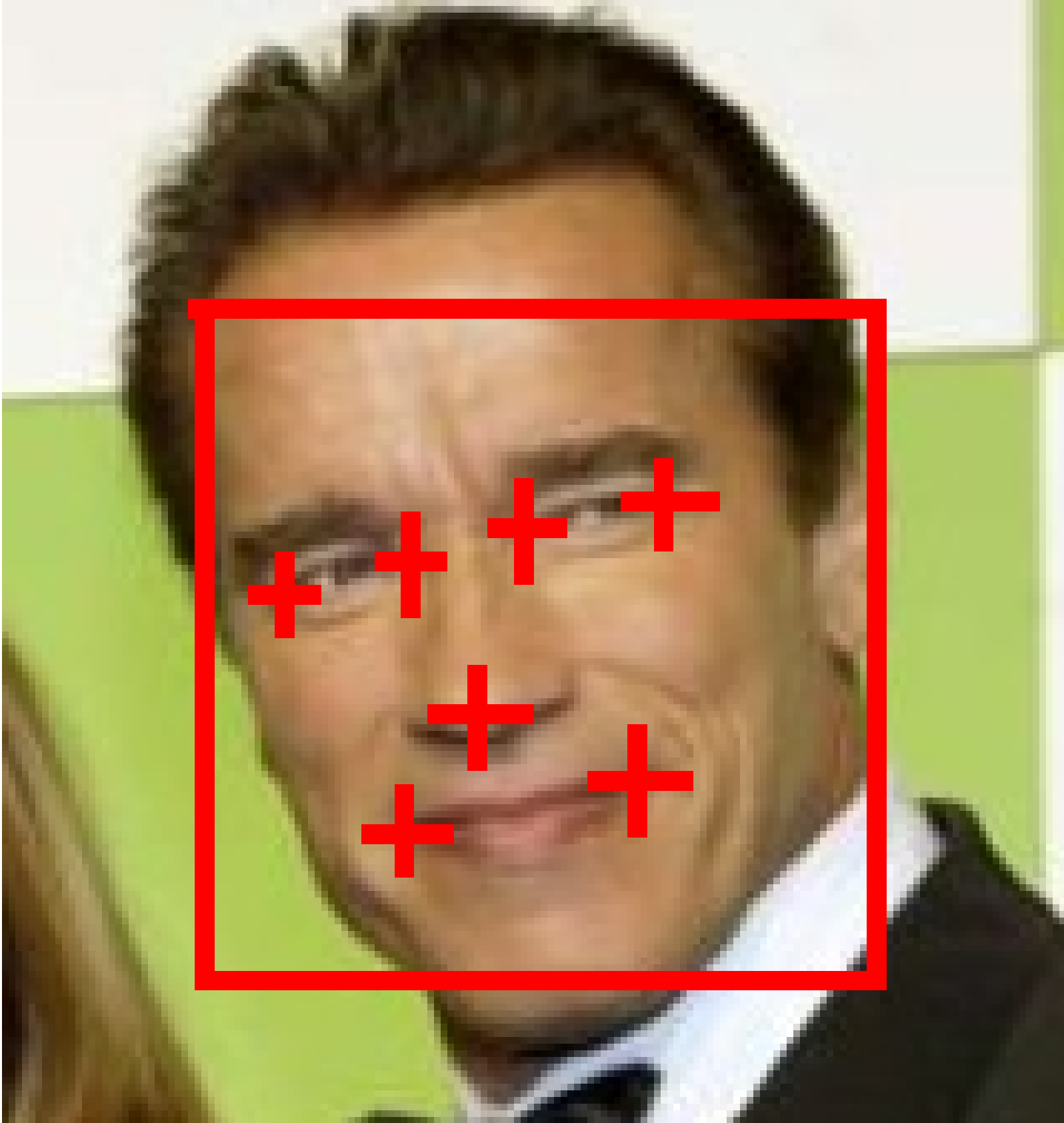
m p

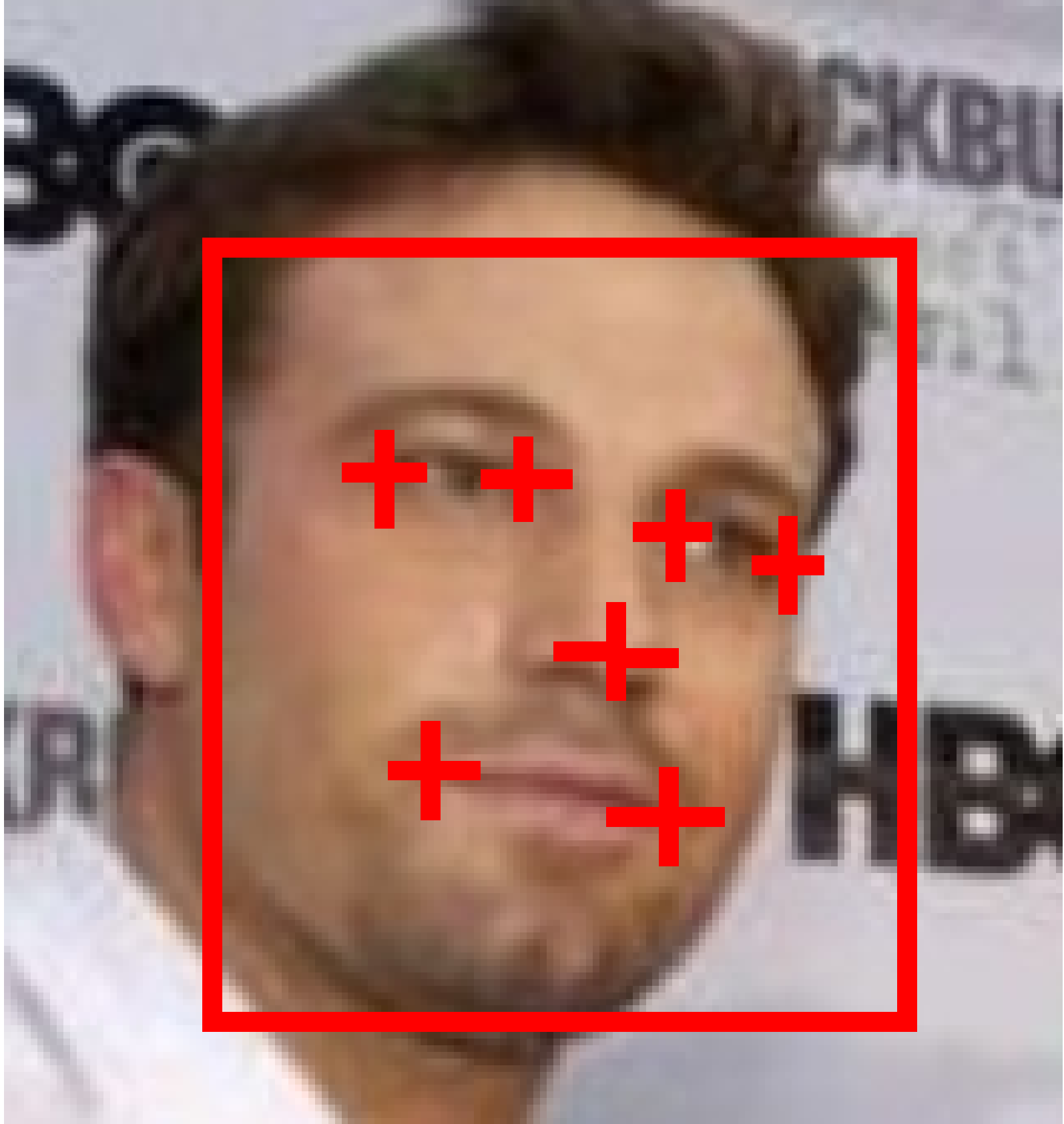


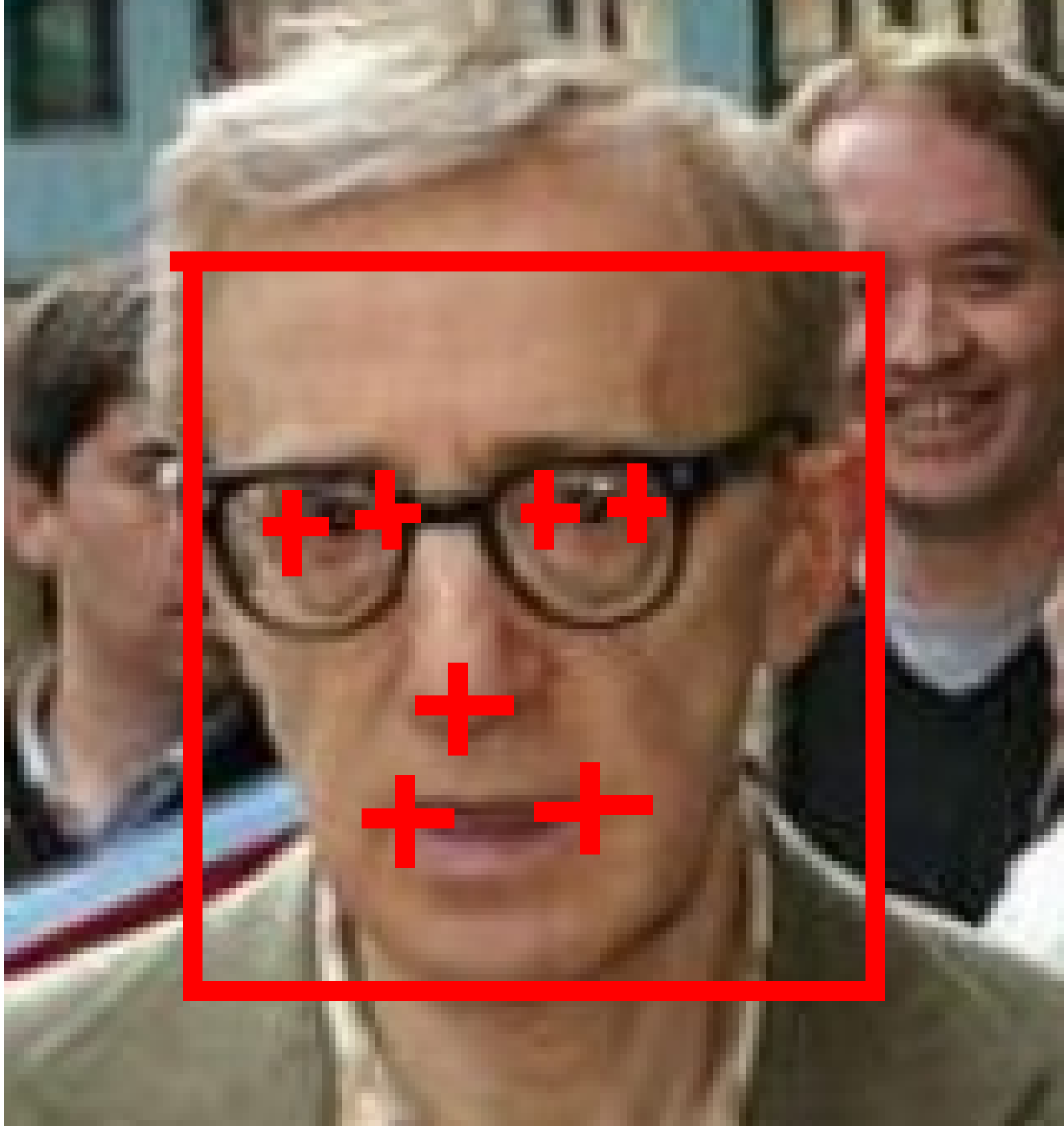


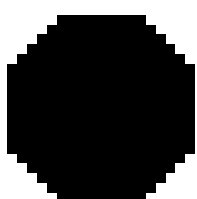
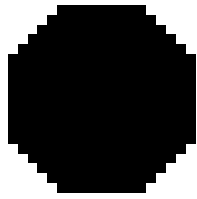
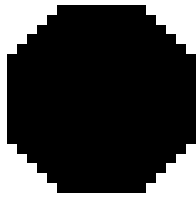




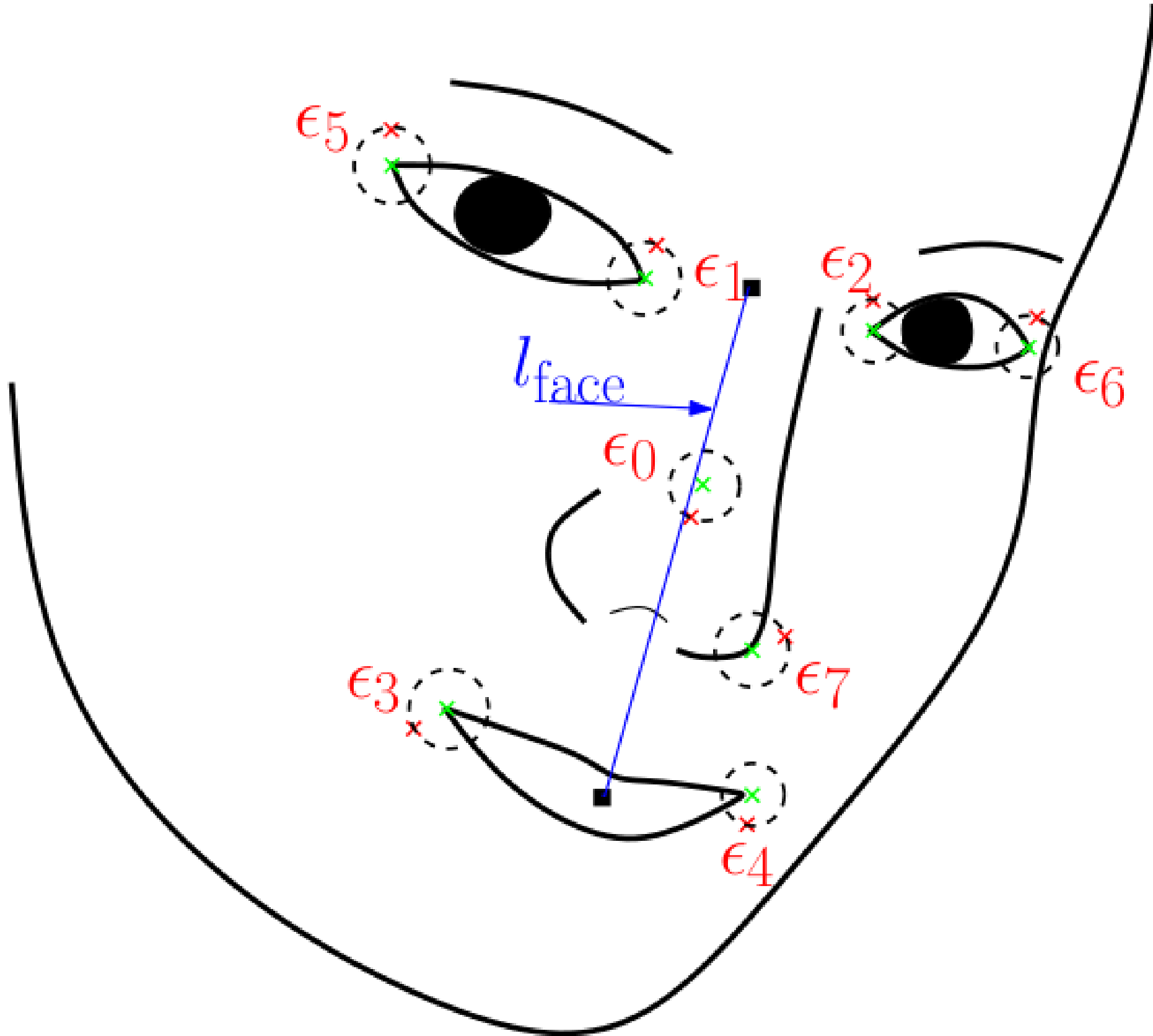




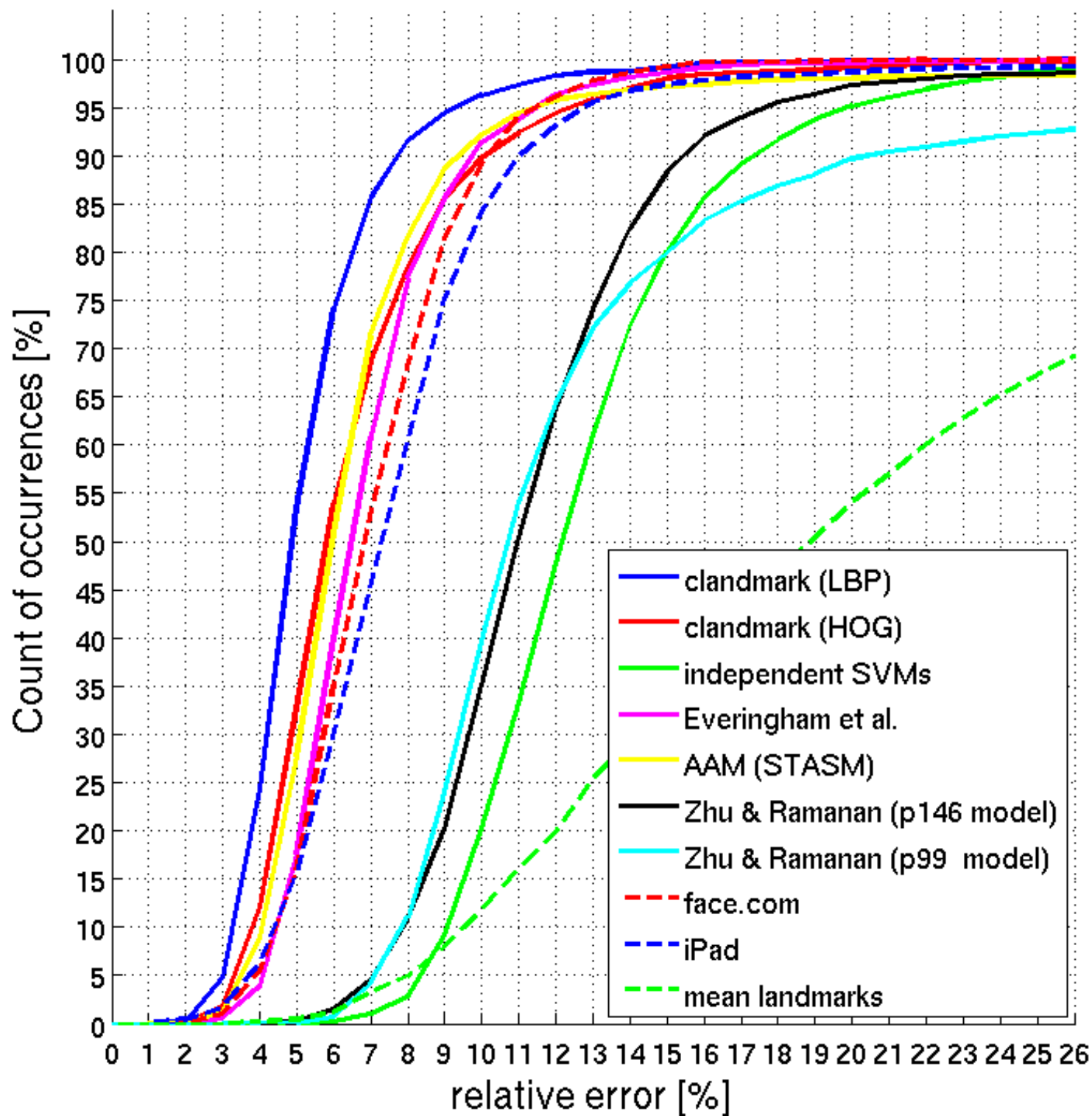


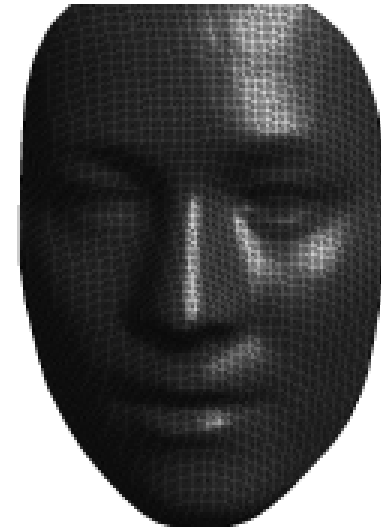
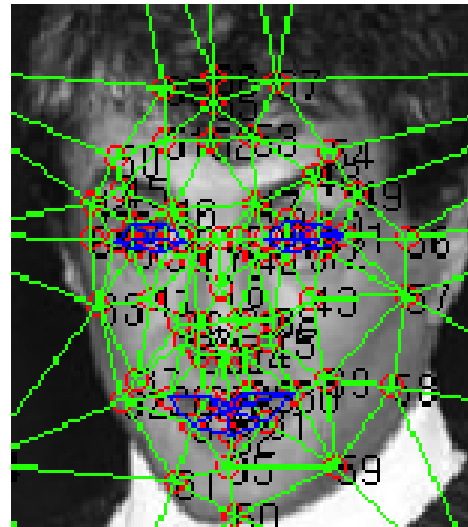




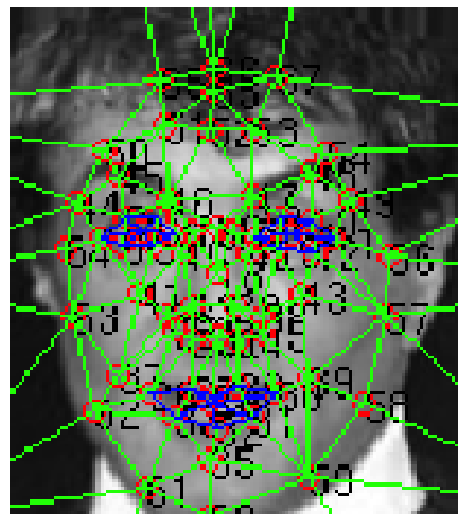
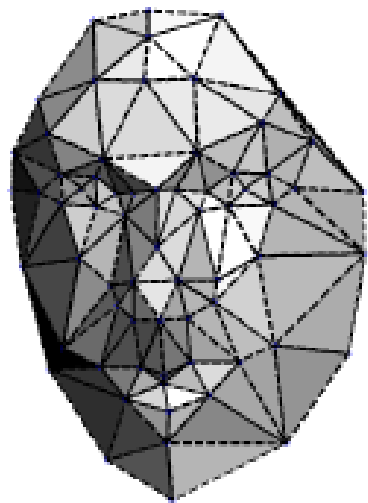


LFW database - Cumulative histogram of mean errors

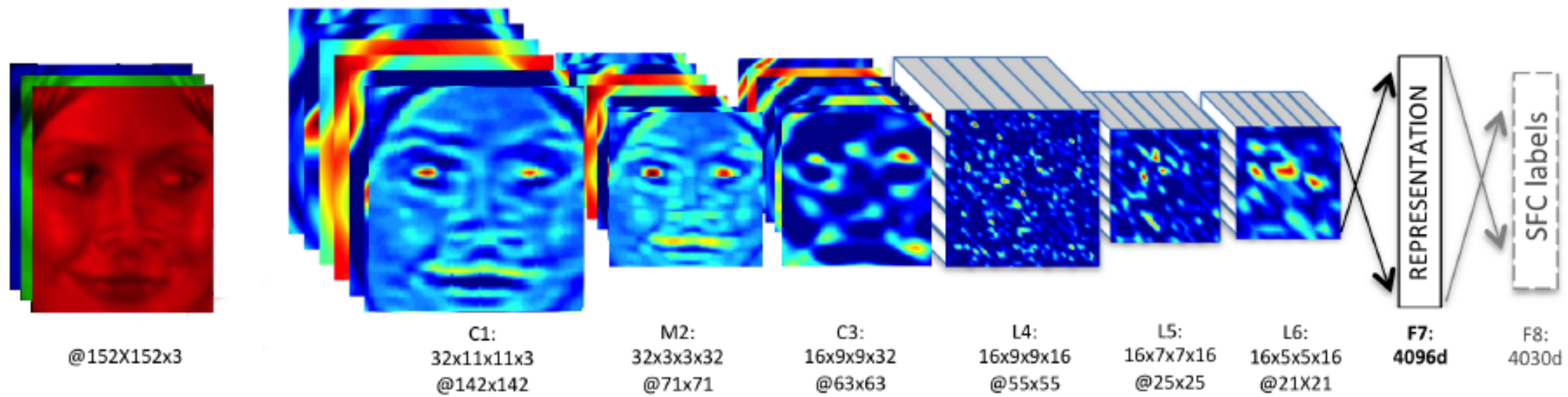






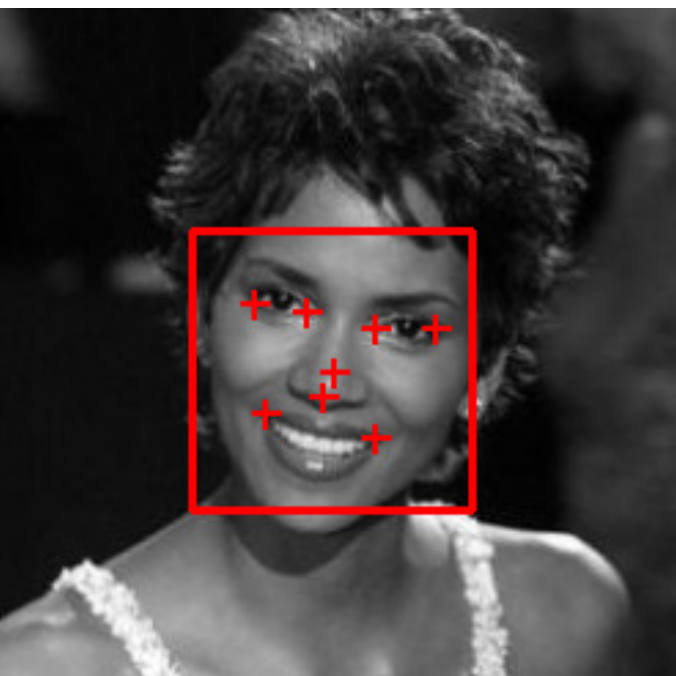


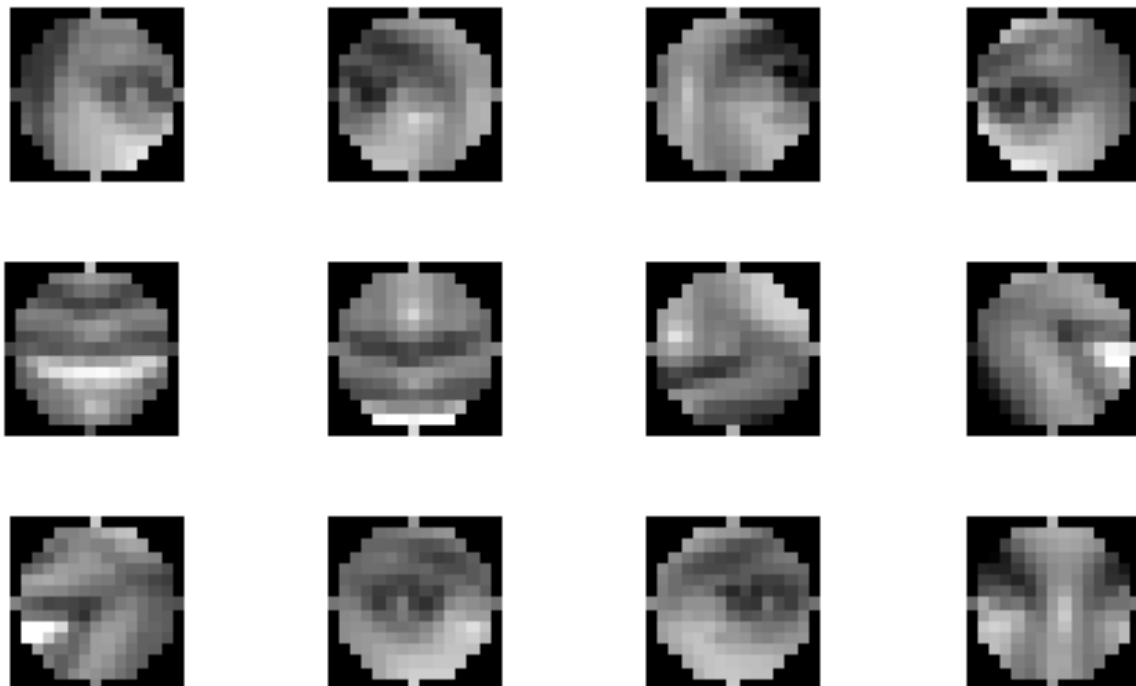
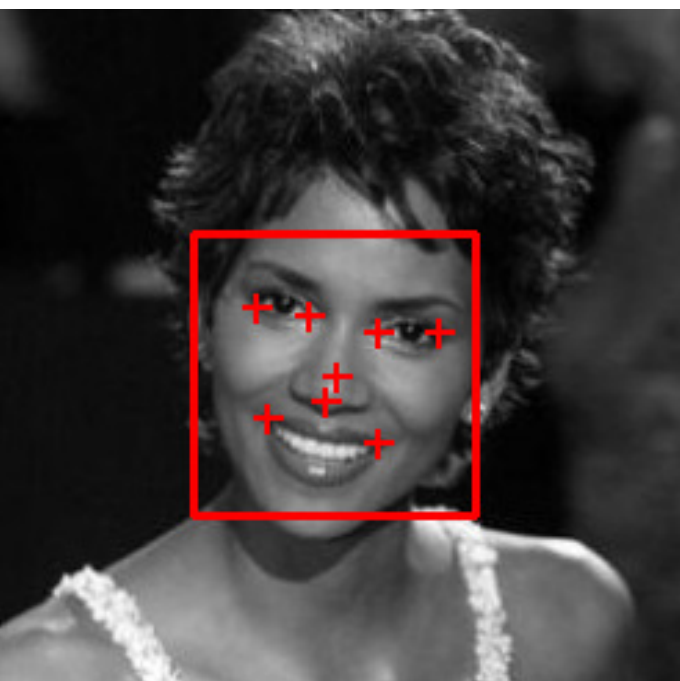










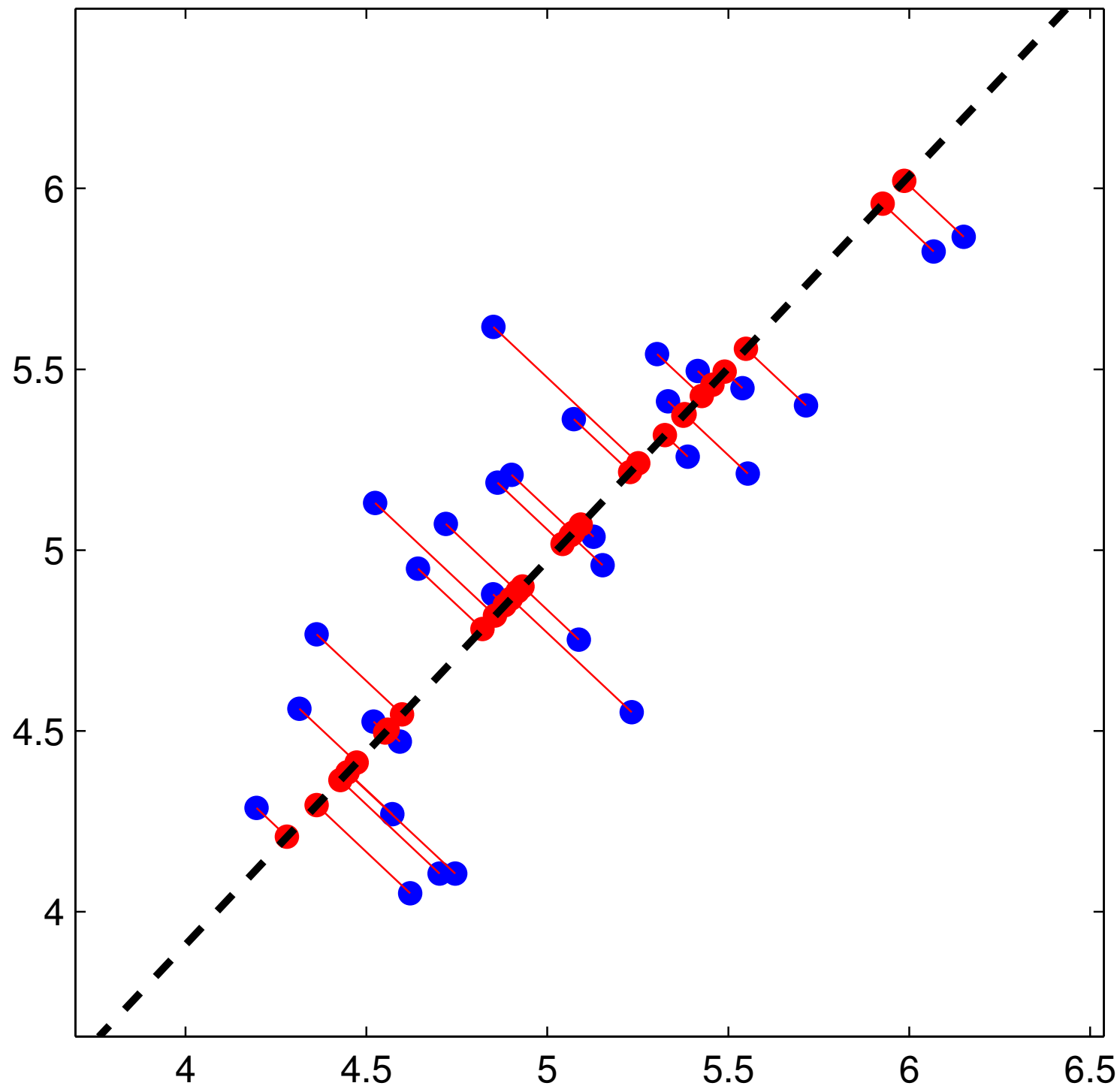


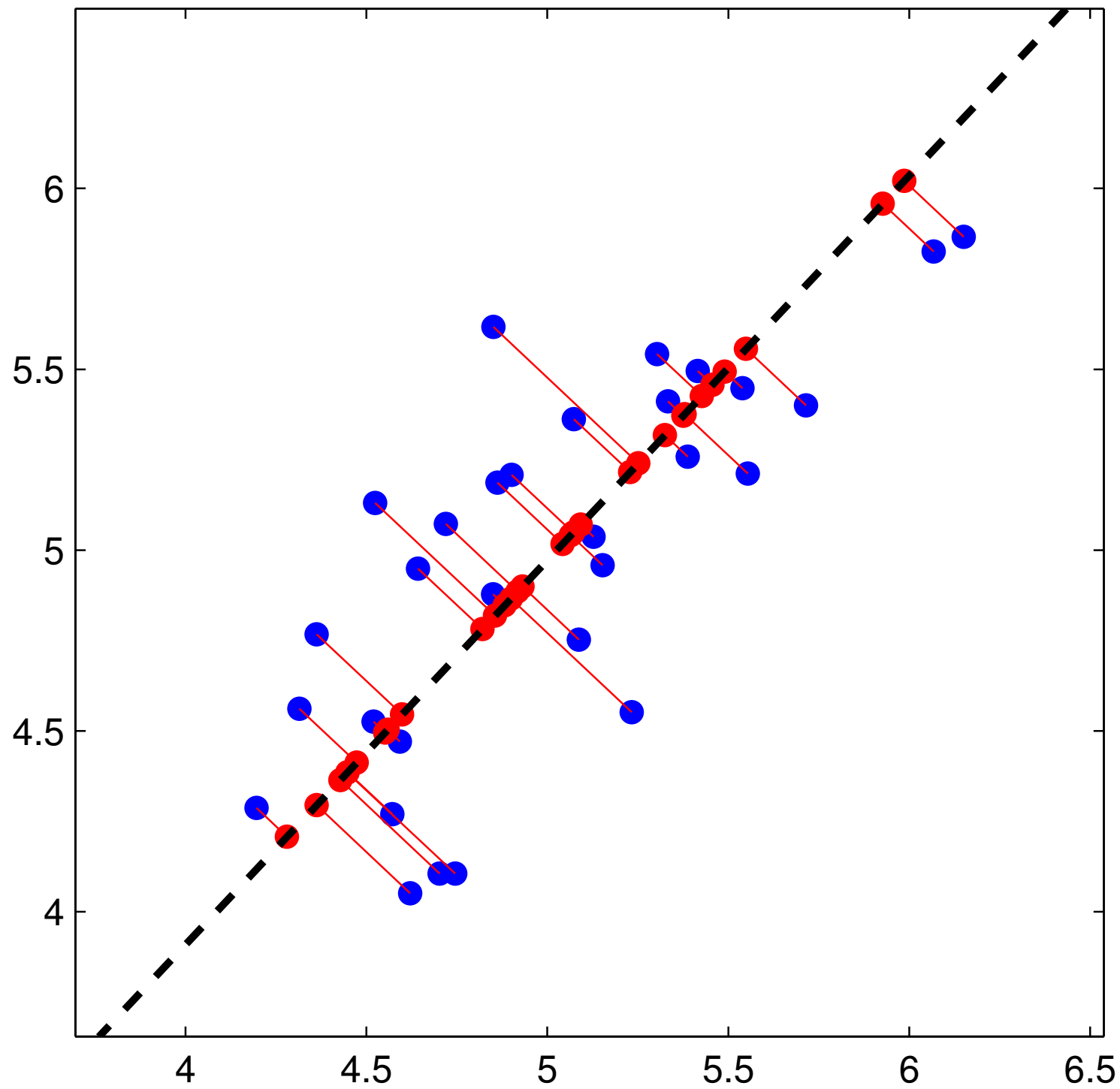
1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16



1
2
3
⋮
16





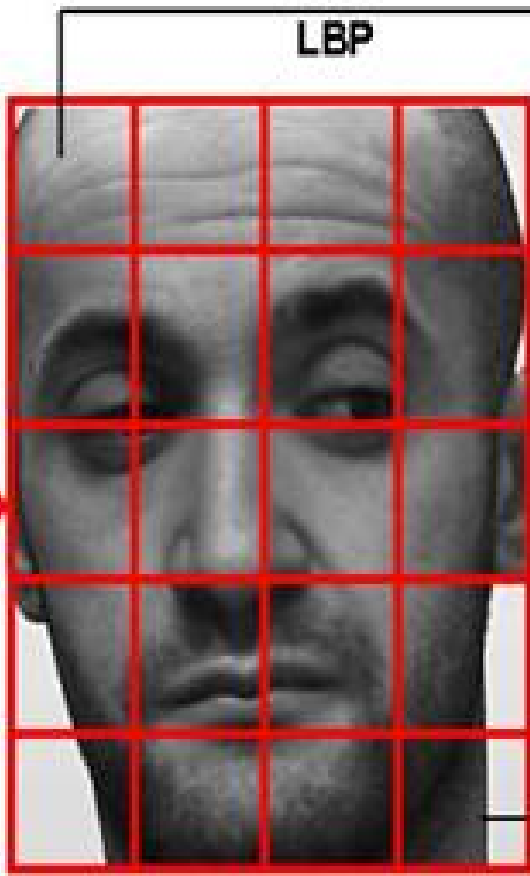




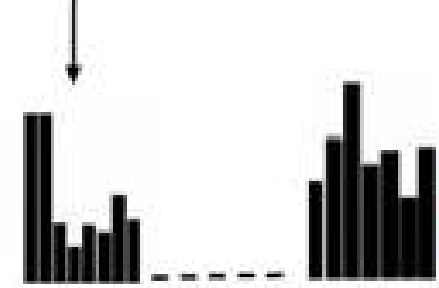
		0	1	2	
		3	c	4	
		5	6	7	



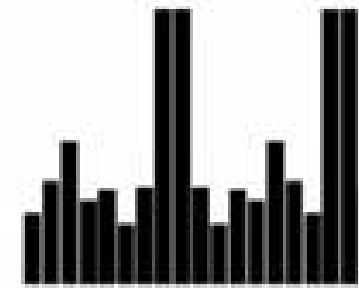
Face image



The face image is divided into blocks



LBP histogram from each block



Feature histogram

