

Rozpoznávání podle obličeje

Tomáš Goldmann, Štěpán Mráček, Martin Drahanský, Jan Váňa
26. 10. 2022

Obsah přednášky

Úvod

Detekce obličeje

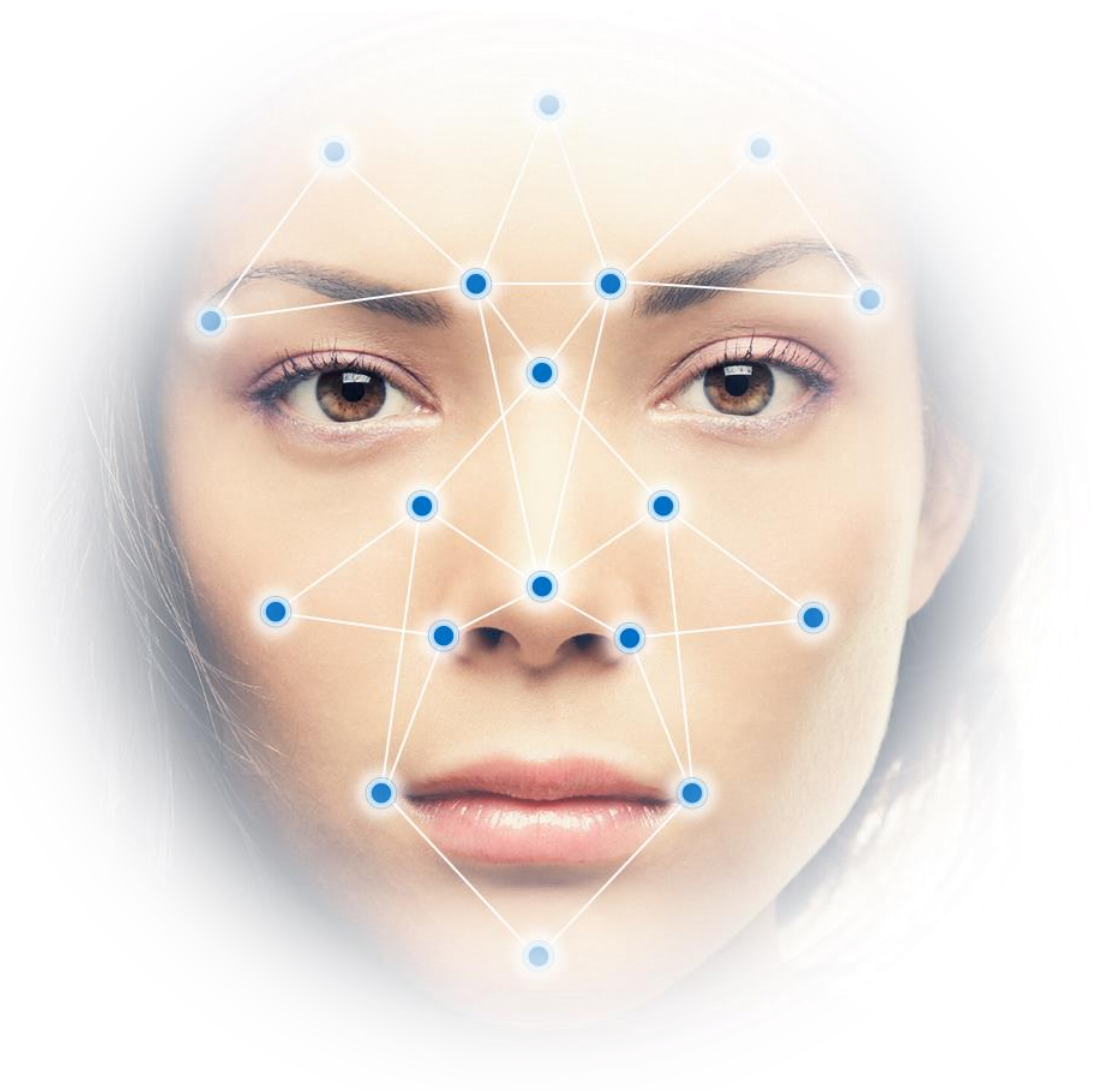
2D obličej

Předzpracování

Extrakce rysů

3D obličej

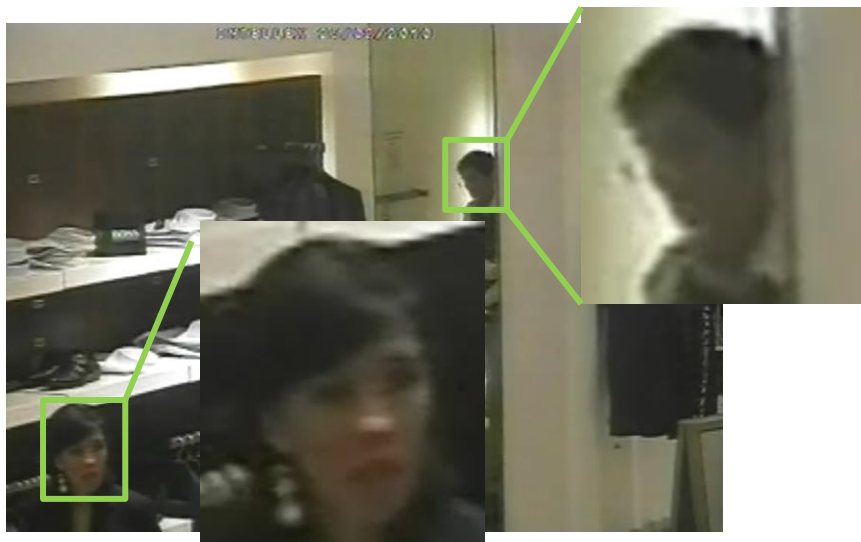
Termosnímký obličeje



Motivace

- Obličej je nejpřirozenější a nejčastěji používanou biometrickou vlastností.
- Výzvy
 - Neřízené rozpoznávání
 - Automatická detekce obličeje
 - Obecné rozpoznání obličeje za různých podmínek
 - Vlivy – různé osvětlení, výrazy obličejů, stárnutí...
- Stejný den, stejné osvětlení: FAR = 2%, FRR = 0,4%
- Stejný den, různé osvětlení: FAR = 2%, FRR = 9%
- Různé dny (> 1 rok): FAR = 2%, FRR = 11% (43%)

Rozpoznávání podle obličeje v praxi - Jak to vypadá v praxi



Motivace

Výhody

Cenová dostupnost

Interpretovatelnost snímaných údajů

Univerzalita aplikací (široké možnosti nasazení)



Nevýhody

- Vnitrotřídní variabilita

- Mimika
- Účes, vousy
- Osvětlení
- Brýle, pokrývky hlavy
- Stárnutí

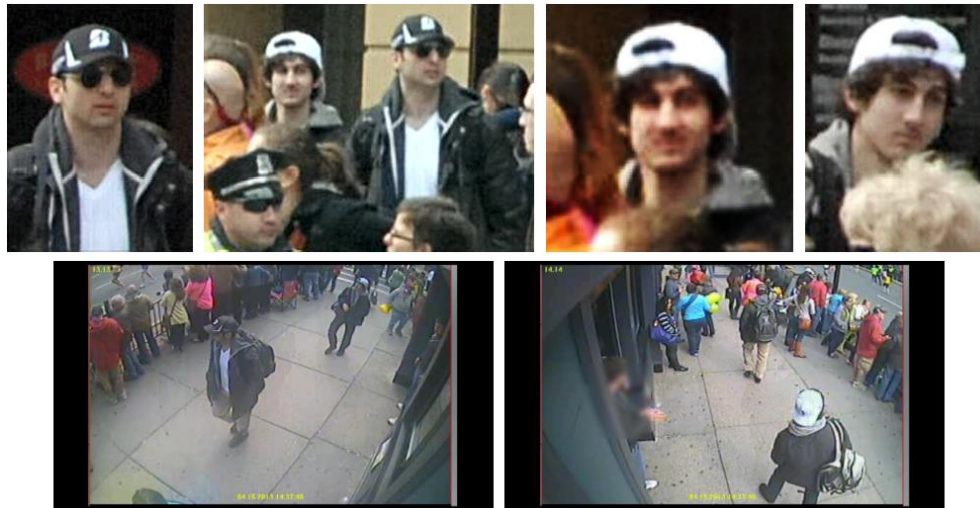
- Mezitřídní variabilita



Rozpoznávání podle obličeje v praxi - Bostonský maraton 2013

Po teroristickém útoku na maraton v Bostnu (2013) se strojově nepodařilo identifikovat útočníky, i přestože FBI měla k dispozici jejich fotky v databázi osob.

**Boston
2013**



**Istanbul
2016**



Zdroj: A Case Study on Unconstrained Facial Recognition Using the Boston Marathon Bombings Suspects

Aplikace #1: přístupové systémy - access control



Aplikace #2: dohledové systémy - surveillance



Time	Alert	Device	Rule	Rule Type	Object	Confidence	Vendor
2011-09-11 00:36:40	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.18164110183716	Ganhetec Global Se
2011-09-11 00:36:29	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.5166020393716	Ganhetec Global Se
2011-09-11 00:36:29	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.0263694908142	Ganhetec Global Se
2011-09-11 00:36:20	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.00791297683716	Ganhetec Global Se
2011-09-11 00:36:11	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.0458979606284	Ganhetec Global Se
2011-09-11 00:36:11	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.0000778655142	Ganhetec Global Se
2011-09-11 00:34:36	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.133789625	Ganhetec Global Se
2011-09-11 00:34:36	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.0009770393716	Ganhetec Global Se
2011-09-11 00:34:33	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	4.03515577316284	Ganhetec Global Se
2011-09-11 00:34:33	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.18164110183716	Ganhetec Global Se
2011-09-11 00:34:33	Ganhetec Alert	AXIS Q6034	Face Identified	Face Identified	JoseCarlosNavarro	3.18164110183716	Ganhetec Global Se

Aplikace #2: dohledové systémy – surveillance



FaceTime	AgeRange	Glass	Sex	FaceRect	FaceUrl
1554079494	0	0	1	{"bottom":662,"left":1718,"right":1788,"top":592}	Capture/190401/0b510d4543f4eaf2333086ef4106530d.png
1554081856	0	0	1	{"bottom":604,"left":904,"right":930,"top":576}	Capture/190401/8be0e2a39dc63c800e7c1582aa6a9965.png
1554081959	0	0	1	{"bottom":600,"left":836,"right":862,"top":574}	Capture/190401/c4fe463f19d7f6fc7cc79470fc09aef.png
1554084555	0	0	0	{"bottom":706,"left":644,"right":684,"top":666}	Capture/190401/6f67457a1fe5ede9b3cc005c9cb562be.png
1554083048	0	0	0	{"bottom":592,"left":592,"right":632,"top":550}	Capture/190401/4889189a0e713ee259708cbc5ea7b857.png
1554084781	0	0	0	{"bottom":894,"left":1204,"right":1288,"top":810}	Capture/190401/50bc6ffa2ad97b34595aaa34412a83ee.png
1554084706	0	1	1	{"bottom":896,"left":1200,"right":1290,"top":806}	Capture/190401/cd0aa0fa747a83f350760fea09c6747c.png
1554082506	0	1	0	{"bottom":704,"left":300,"right":348,"top":656}	Capture/190401/e5a74325b50589e2b441460b2d325061.png
1554083066	0	0	0	{"bottom":588,"left":576,"right":612,"top":550}	Capture/190401/fb5c6de8248ae5b2324cf48883a1039d.png
1554083776	0	1	0	{"bottom":870,"left":1174,"right":1260,"top":784}	Capture/190401/2ef061f0ef964a4bb06143731a8d113f.png
1554083181	0	1	0	{"bottom":868,"left":1192,"right":1276,"top":784}	Capture/190401/3540ec6694431e57cd8dd5f96653b94e.png
1554084628	0	1	0	{"bottom":890,"left":1206,"right":1286,"top":808}	Capture/190401/9c12c5f8848c787503b79a2419d35124.png
1554084976	0	1	0	{"bottom":882,"left":1212,"right":1286,"top":808}	Capture/190401/4babab72e6c4990b3d000dfa03d47e.png

Aplikace #2: dohledové systémy – surveillance



<https://ipvm.com/reports/hikvision-uyghur>

Dahua History of Race/Ethnicity Analytics

In November 2020, Dahua ([inadvertently](#)) went viral on Twitter after an engineer posted lines of code from Dahua's SDK showing "EM_NATION_TYPE_UYGUR" next to "维族(新疆)" which means "Uyghur (Xinjiang)". Dahua's SDK at the time also mentioned three distinct 'races' by skin color - "black", "white", and "yellow":

```
// 种族类型
public static class EM_RACE_TYPE extends MyStructure
{
    public static final int EM_RACE_UNKNOWN = 0;           // 未知
    public static final int EM_RACE_NODISTI = 1;          // 未识别
    public static final int EM_RACE_YELLOW = 2;           // 黄种人
    public static final int EM_RACE_BLACK = 3;             // 黑人
    public static final int EM_RACE_WHITE = 4;             // 白人
}
```

In February 2021, [IPVM and The LA Times](#) revealed Dahua technical documents for PRC police video surveillance systems which included "real-time Uyghur warnings/维族人员实时预警":

人脸识别法	维族人员实时预警 (Real-time Uyghur warnings)
	1. web操作, 人脸系统-研判分析, 配置维族人员模型 2. 客户端, 人脸识别-数据模型-模型查询, 选择维族人员模型, 查询到对应数据 3. 前端需要支持支持上报警告属性

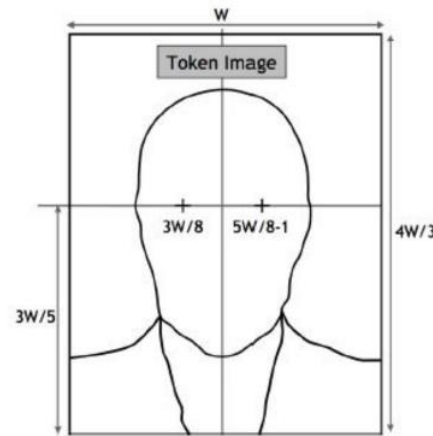
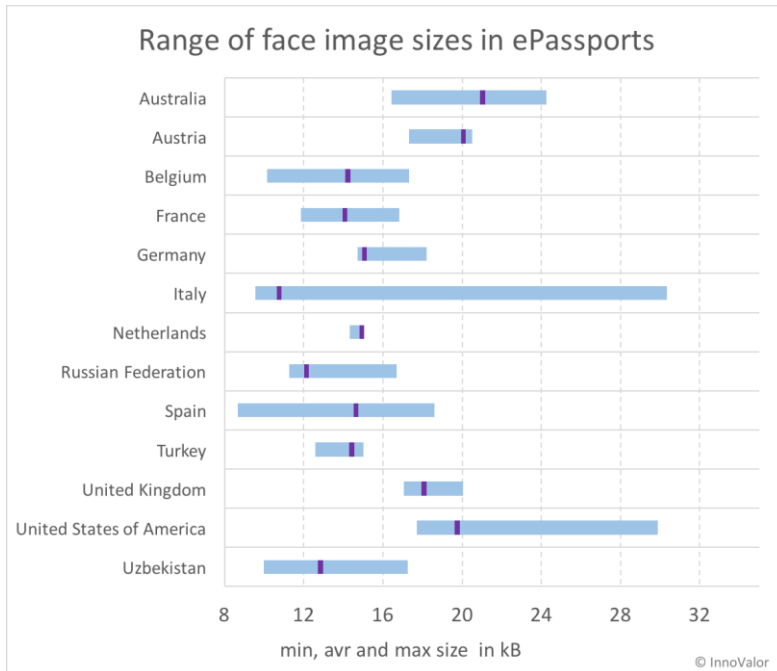
The video below explains how the "Uyghur warnings" worked:

<https://ipvm.com/reports/dahua-race-analytics>

Aplikace #3: biometrické pasy (jenom z pohledu rozpoznávání podle obličeje)

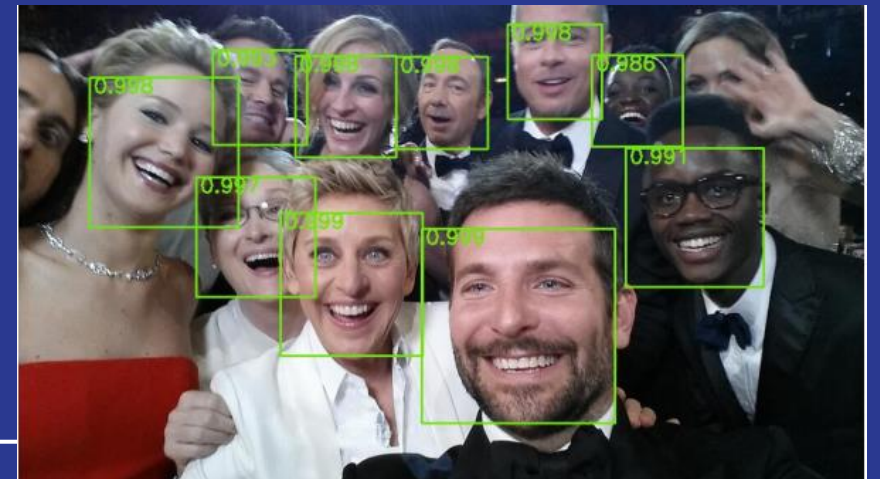
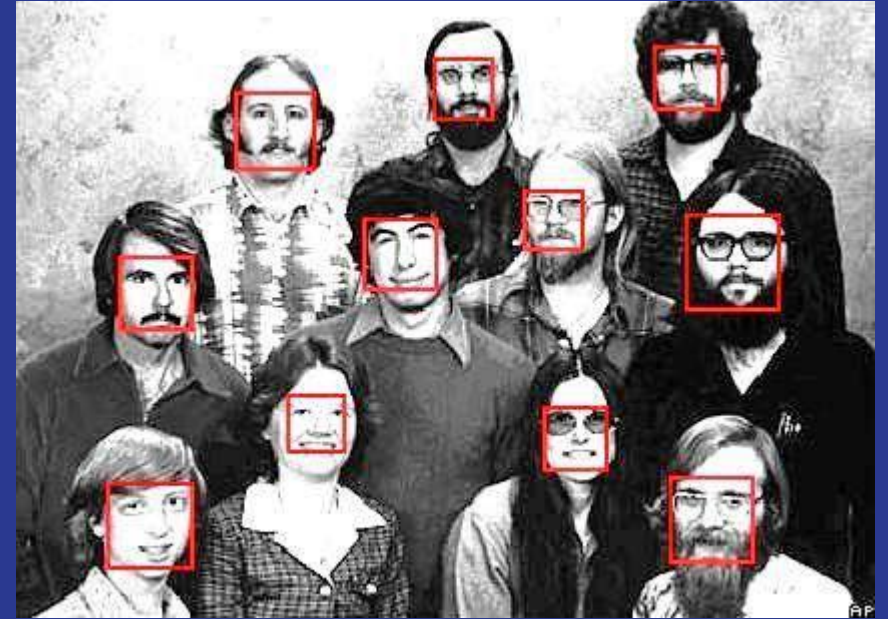
Vyfocení podle ISO/IEC 19794-5

Rozměr fotky není určený normou a každý stát si jej může určit sám. Omezené místo – min. 12kB, optimálně 15-20kB.



Canonical Portrait Photograph,
as standardized in ISO/IEC
19794-5 (now superseded by
ISO/IEC 39794-5).

Detekce obličejů



Detekce obličeje

- Jedná se o první krok ve zpracování obličejů.
- Obličeje jsou nasnímány za různých osvětlení.
- Obličeje se liší v barvě, pozici, rozměrech, orientaci, 3D pozici, výrazu obličeje atd.
- Problémem je větší množství obličejů s barevným pozadím.



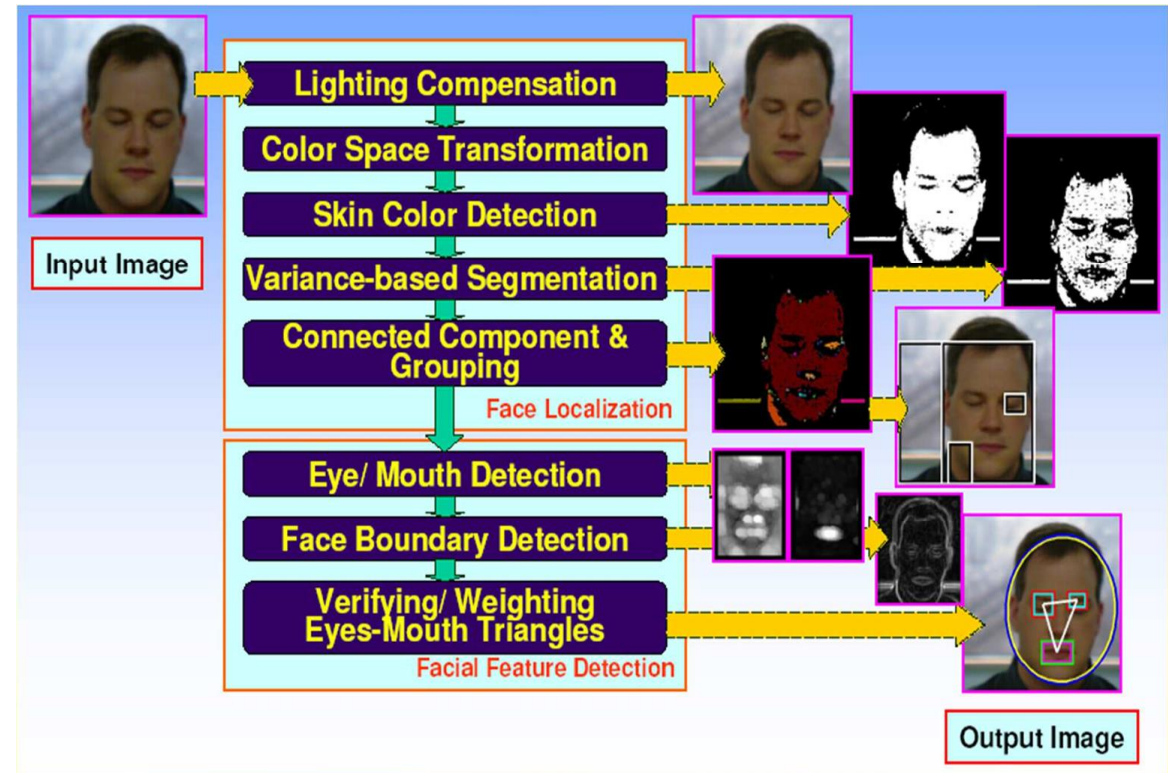
Fotografie z Pražského metra

Detekce obličeje

1. Analýza barevného prostoru (Algoritmus I).
2. Kaskáda klasifikátorů, Adaboost, Haarovy vlnky (Algoritmus II).
3. Histogram orientovaných gradientů + SVM (Algoritmus III).
4. Detektory obličeje založené na neuronových sítích (například https://docs.openvino.ai/2019_R1/face_detection_adas_binary_0001_description_face_detection_adas_binary_0001.html)

Algoritmus I - analýza barevného prostoru

- Kompenzace osvětlení
- Detekce barvy kůže
- Segmentace
- Detekce úst
- Detekce očí



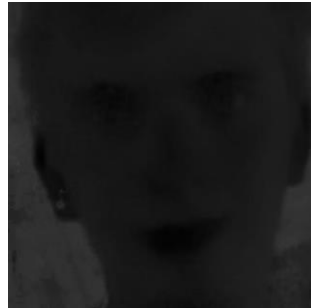
Barevné prostory HSV a YCbCr

- H – Hue, S – Saturation, V – Value
- Y - Luma – intenzita, C_B - Blue difference chroma component, C_R - Red difference chroma component

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \quad \begin{array}{l} Y \in [0, 255] \\ C_b \in [0, 255] \\ C_r \in [0, 255] \end{array}$$



RGB



Hue



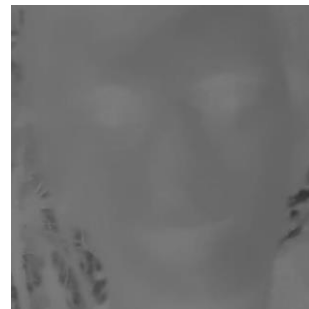
Saturation



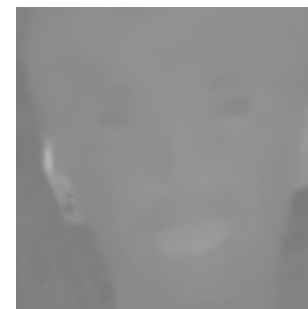
Value



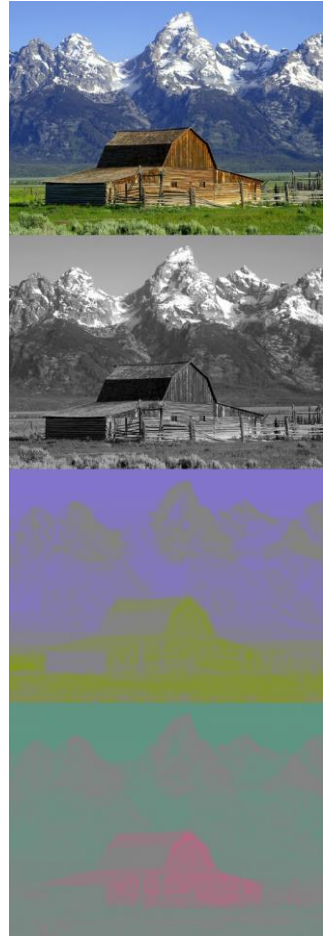
Luma



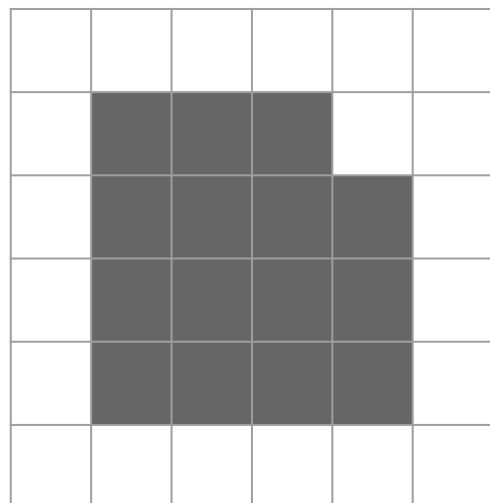
Cb



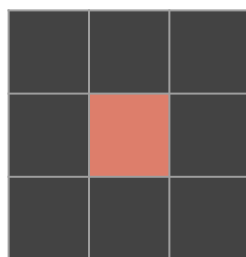
Cr



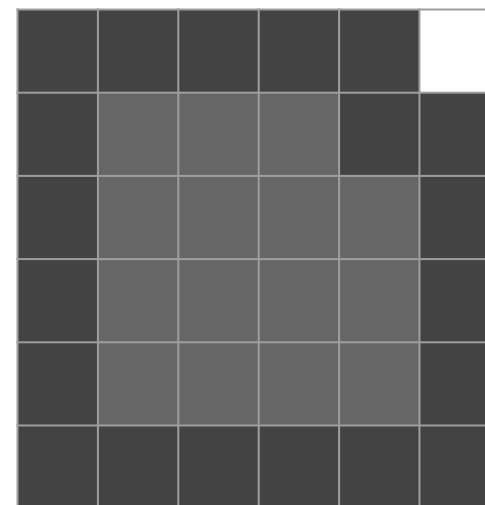
Morfologické operace



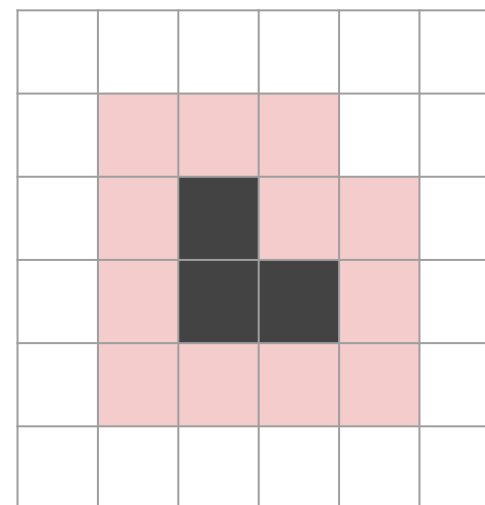
Vstupní obrázek - A



Jádro
Structuring element
Kernel
B

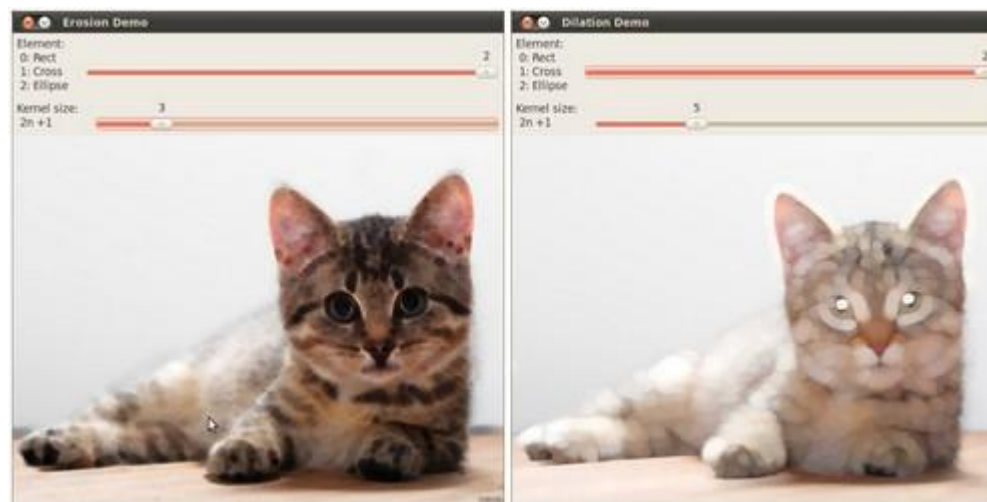


Dilatace: $A \oplus B$

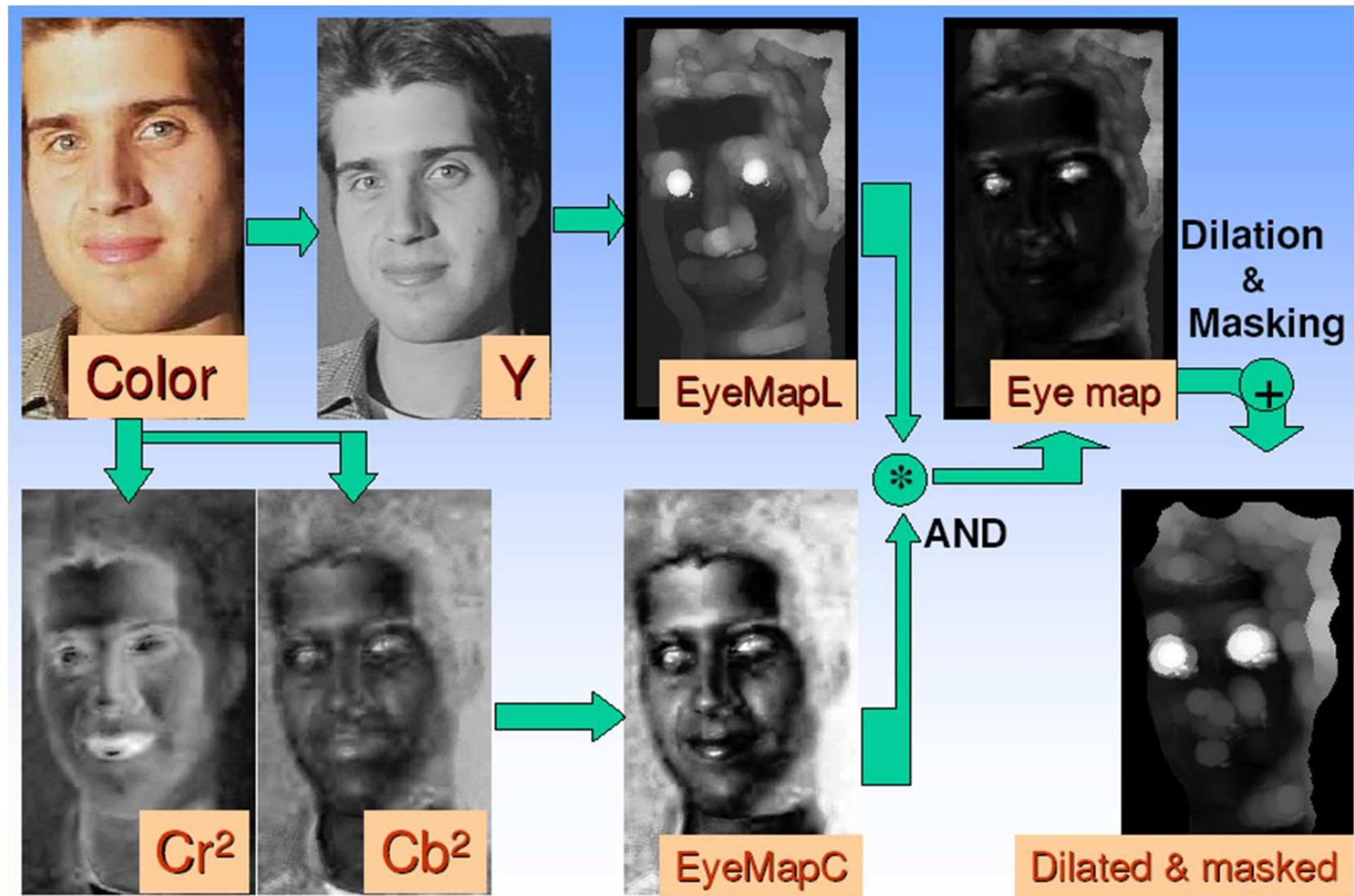


Eroze: $A \ominus B$

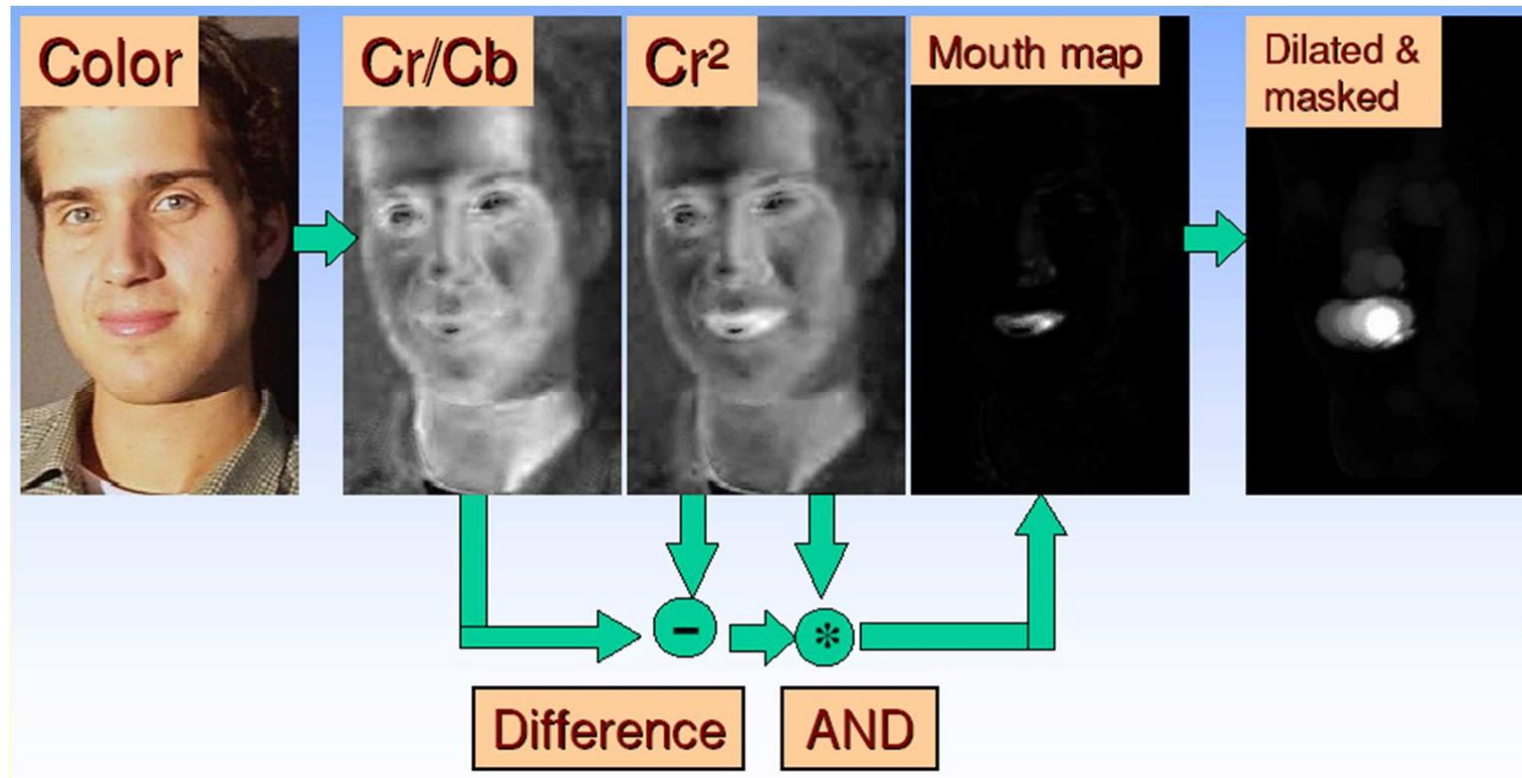
Morfologické operace



Algoritmus I - detekce očí

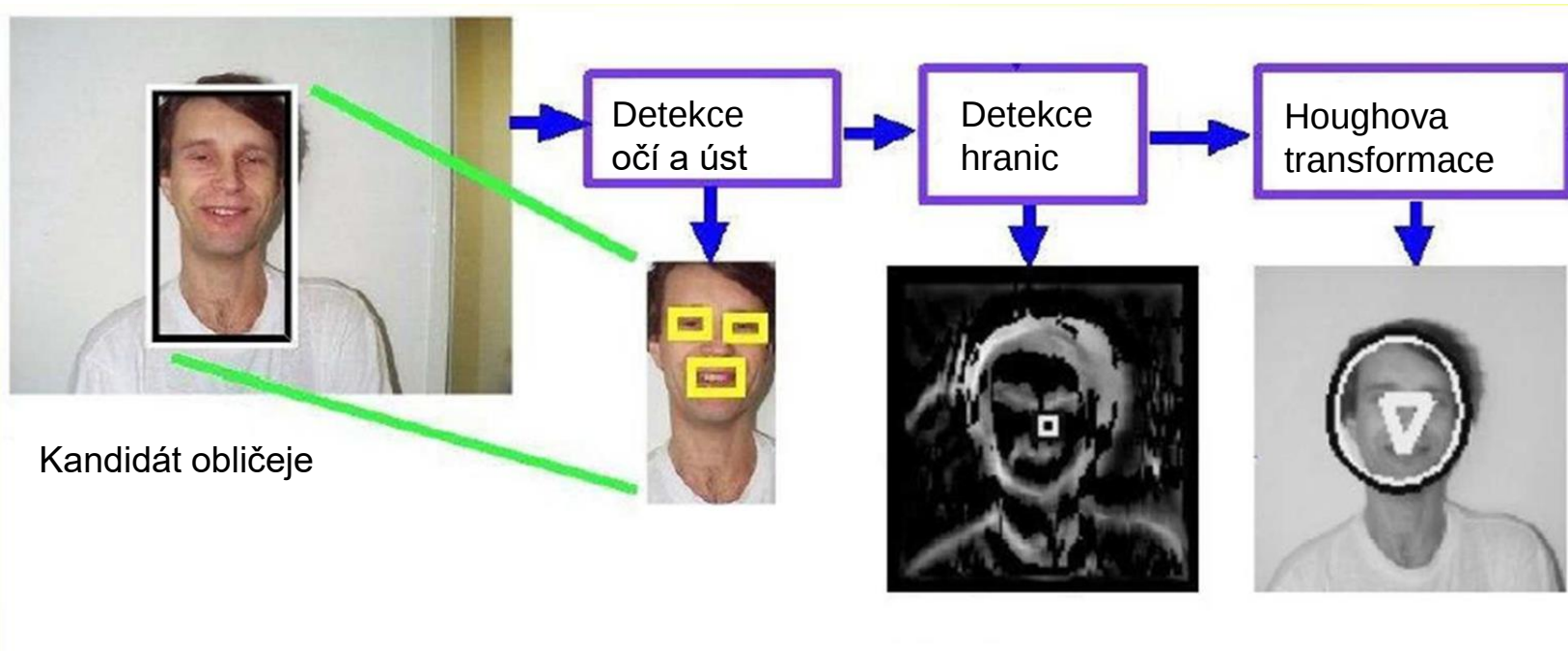


Algoritmus I - detekce úst



Algoritmus I - detekce hranice obličeje

Hranice obličeje (výpočet založen na velikosti samotného obličeje a orientacích gradientů).



Algoritmus II - strojové učení - AdaBoost

Vstup: Množina slabých klasifikátorů, pozitivní vs. negativní příklady

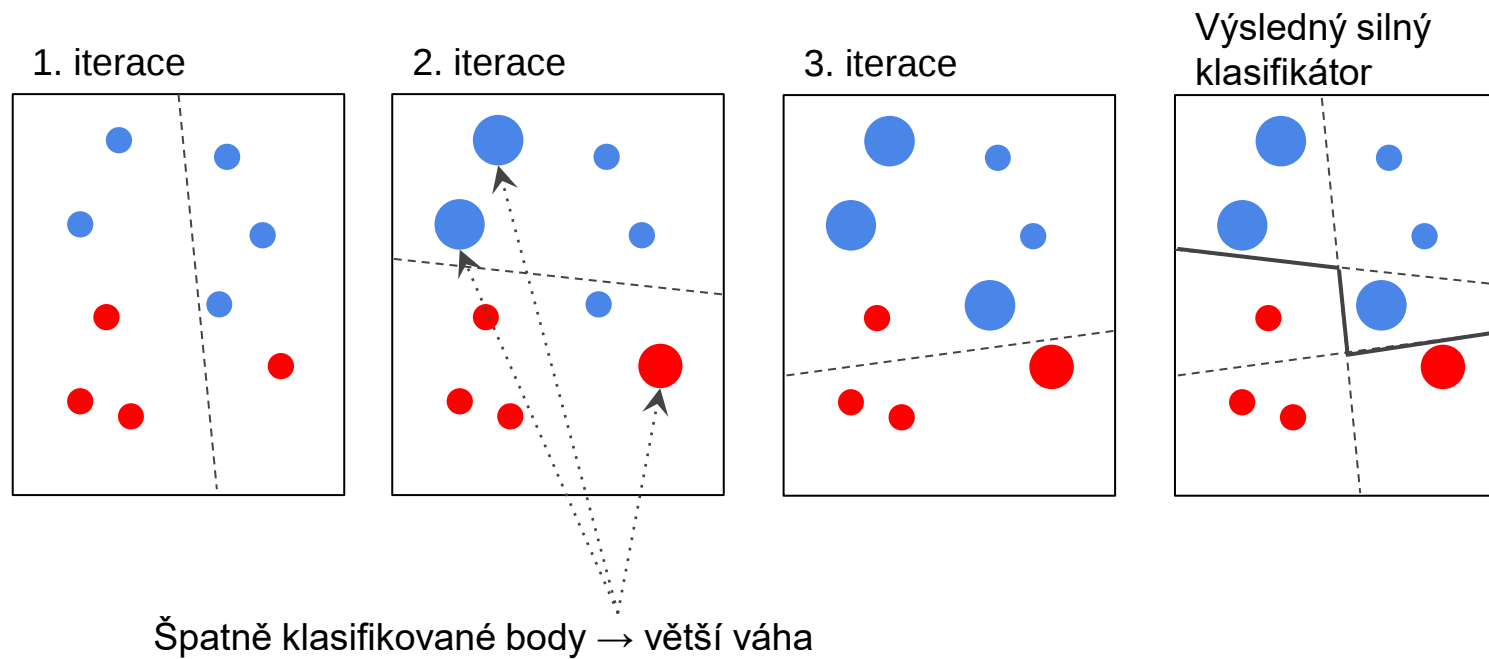
Výstup: Silný klasifikátor

Postup:

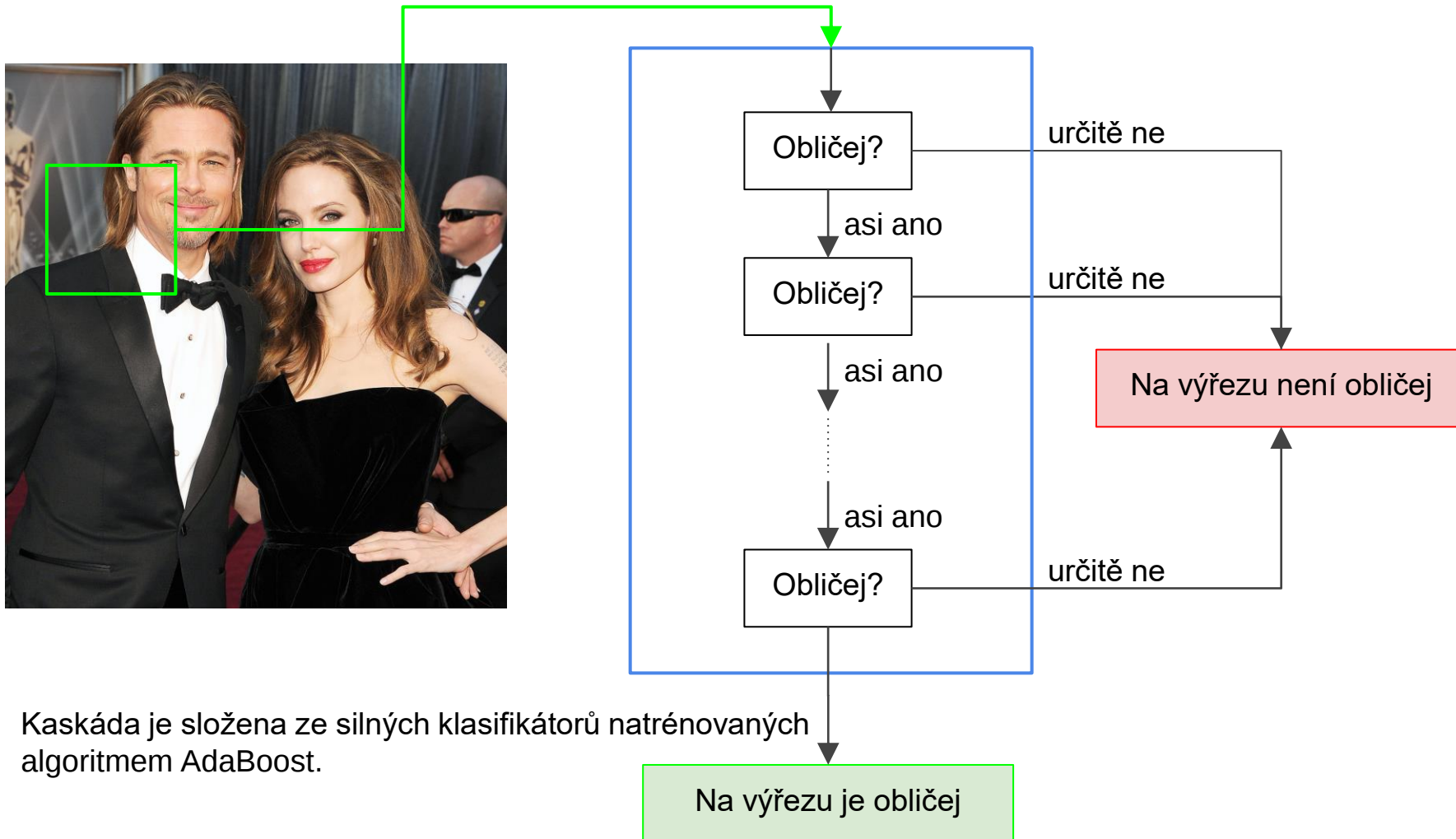
1. Nastav váhy trénovacích dat na výchozí hodnotu.
2. Z množiny slabých klasifikátorů vyber ten, který dosahuje nejlepších výsledků na trénovacích datech (v závislosti na vahách).
3. Vypočítej koeficient pro tento klasifikátor na základě jeho úspěšnosti.
4. Uprav váhy trénovacích dat (váha roste pro data, která jsou špatně klasifikována a naopak)
5. Pokračuj krokem 2, dokud není výsledný klasifikátor dostatečně přesný.

Viola-Jones detektor (2001) = AdaBoost + Haarovy příznaky + kaskáda klasifikátorů

Algoritmus II - strojové učení - AdaBoost



Algoritmus II - kaskáda klasifikátorů

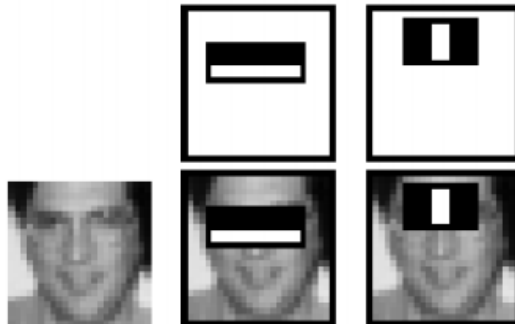


Algoritmus II - slabý klasifikátor

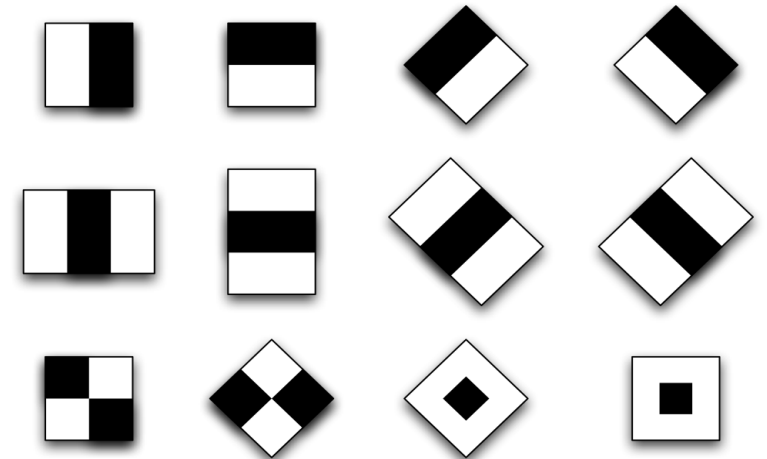
2D Haarova vlnka.

Přiloží se k obrazu a vypočítá se rozdíl součtů intenzit pixelů obrazu pod černou nebo bílou oblastí vlnky.

- Kladná odezva → jedná se o obličej.
- Záporná odezva → není obličej.



$$f(x, y) = \sum_i p_b(i) - \sum_i p_w(i)$$



Algoritmus II - Výpočet odezvy na Haarovu vlnku

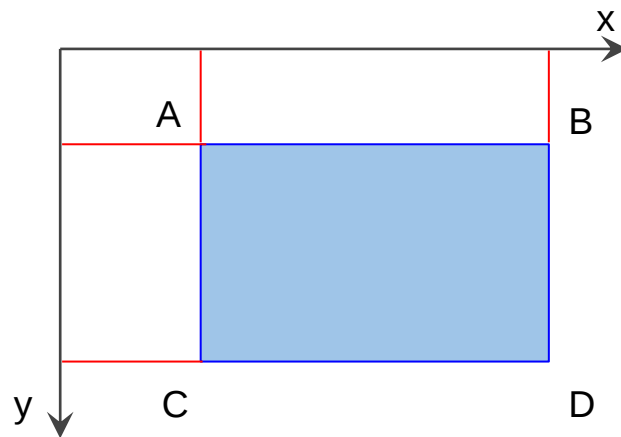
Opakovaný výpočet sumy intenzit pixelů v určitém regionu obrázku je poměrně náročný.

Je proto vhodné obrázek $i(x,y)$ převést do tzv. **Integrální** podoby:
$$I_{\Sigma}(x,y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i(x',y')$$

Výpočet součtu intenzit pixelů v obdélníku daným body A,B,C,D je pak dán výrazem:

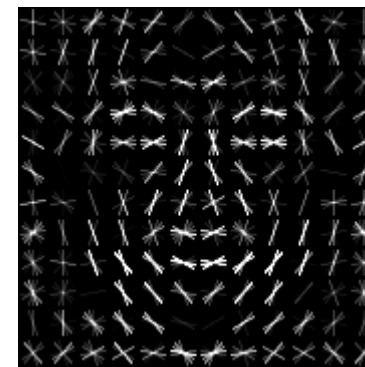
$$I(D)-I(B)-I(C)+I(A)$$

kde $I(A), I(B), I(C)$ a $I(D)$ jsou hodnoty integrálního obrázku v daných bodech



Algoritmus III - Histogram orientovaných gradientů + Support Vector Machines

1. Vstupnímu obrázku vypočítáme gradient („Jakým směrem a jak moc se mění intenzita pixelů“).
2. Dostaneme vektorové pole.
3. Pole rozdělíme do pravidelné mřížky.
4. V každé buňce mřížky vypočítáme histogram orientací gradientů.
5. Histogramy spojíme dohromady - tvoří vektor rysů.
6. Tento vektor klasifikujeme pomocí SVM klasifikátoru.



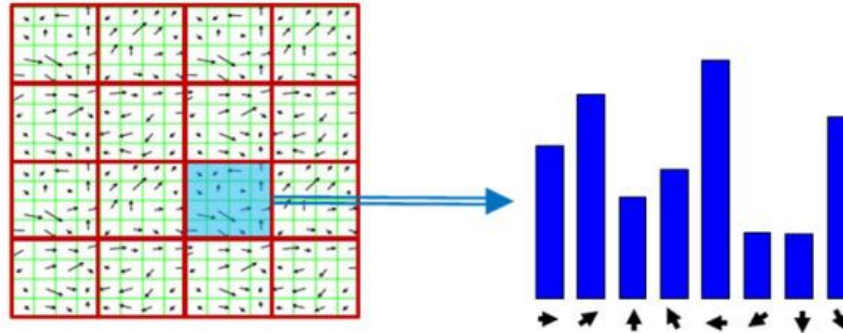
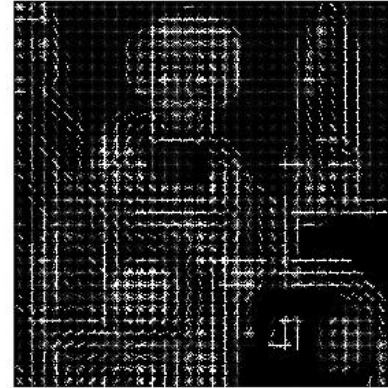
Výhody oproti AdaBoost: menší množství chybných detekcí při stejné přesnosti; pro natrénování stačí menší množství příkladů; méně parametrů, které je třeba nastavit

Algoritmus III - Histogram orientovaných gradientů

Input image



Histogram of Oriented Gradients



Algoritmus III - Support Vector Machines (SVM)

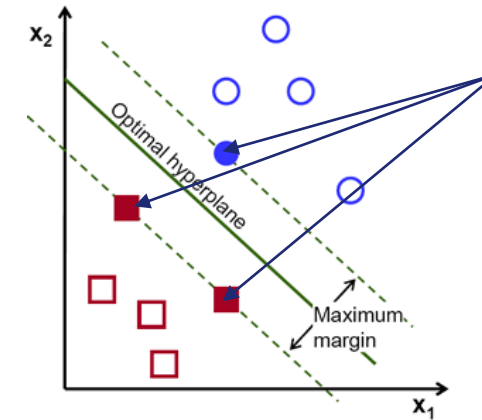
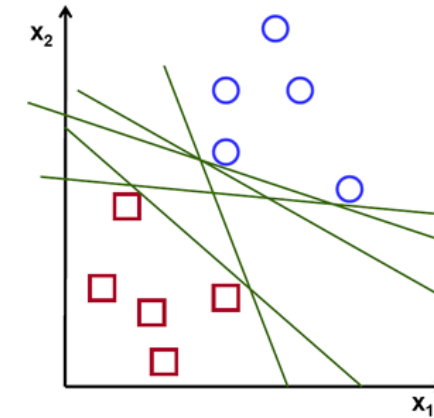
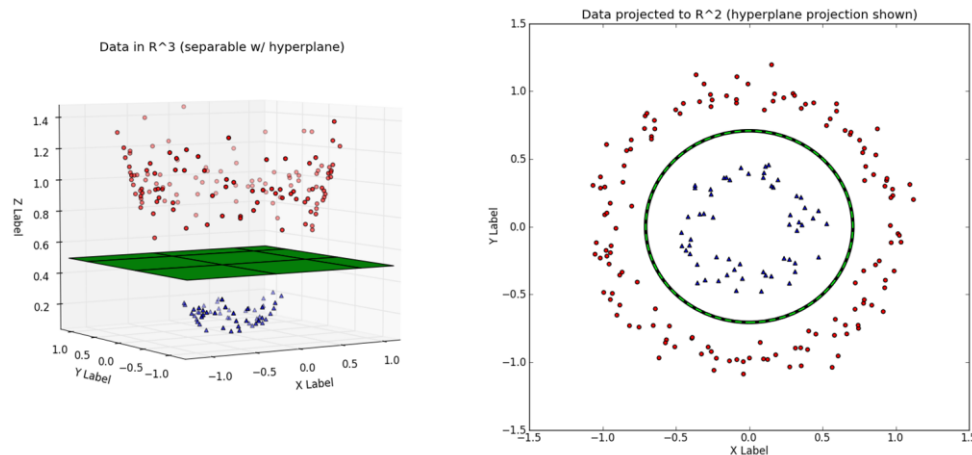
Binární klasifikátor.

Algoritmus s učitelem (Supervised learning).

Diskriminativní model.

Hyperrovina, která „nejlépe“ rozděluje data.

Může být lineární nebo pomocí tzv. kernelů i nelineární



Podpůrné vektory

Algoritmy pro detekci obličeje porovnání



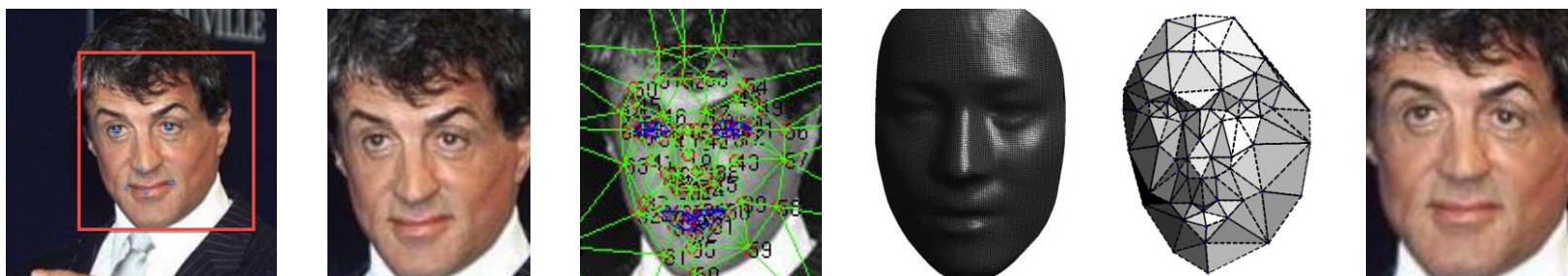
Rozpoznávání 2D obličeje

Předzpracování obrazu



Předzpracování obrazu

Je vhodné mít obličaje od jedné osoby vzájemně co nejpodobnější, proto se používá **normalizace**.



Příklad algoritmu

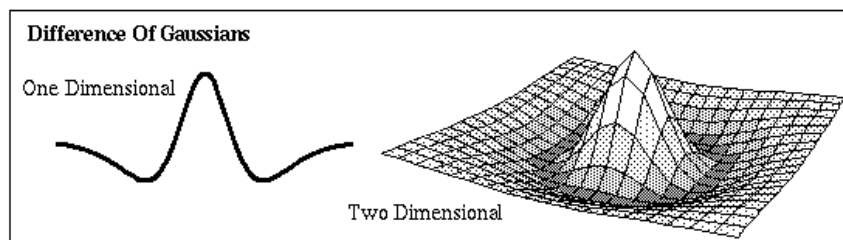
DeepFace, Facebook, Taigman 2014:

<https://research.facebook.com/publications/480567225376225/deepface-closing-the-gap-to-human-level-performance-in-face-verification/>

1. Vstupní obrázek (detekuje se 6 bodů).
2. Jednoduché oříznutí a zarovnání.
3. Detekce 64 bodů.
4. Referenční 3D model.
5. Odhadnutí rotace obličeje pomocí detekovaných bodů a referenčního modelu.
6. Zarovnaný obrázek.

Předzpracování obrazu

Tan a Tiggs 2009: Difference of Gaussians - Vypočítáme dva různě Gaussovsky rozostřené obrázky a odečteme je od sebe. **Pásmová propust'.**



Vstupní
obrázek

Gamma
korece

Rozdíl
rozostřených
obrázků

Ekvalizace
histogramu

Rozpoznávání 2D obličeje

Extrakce rysů



[0.123; -0.081; 12.468; 1.201; -15.787; 0.235]

Přehled metod

Metriky obličeje

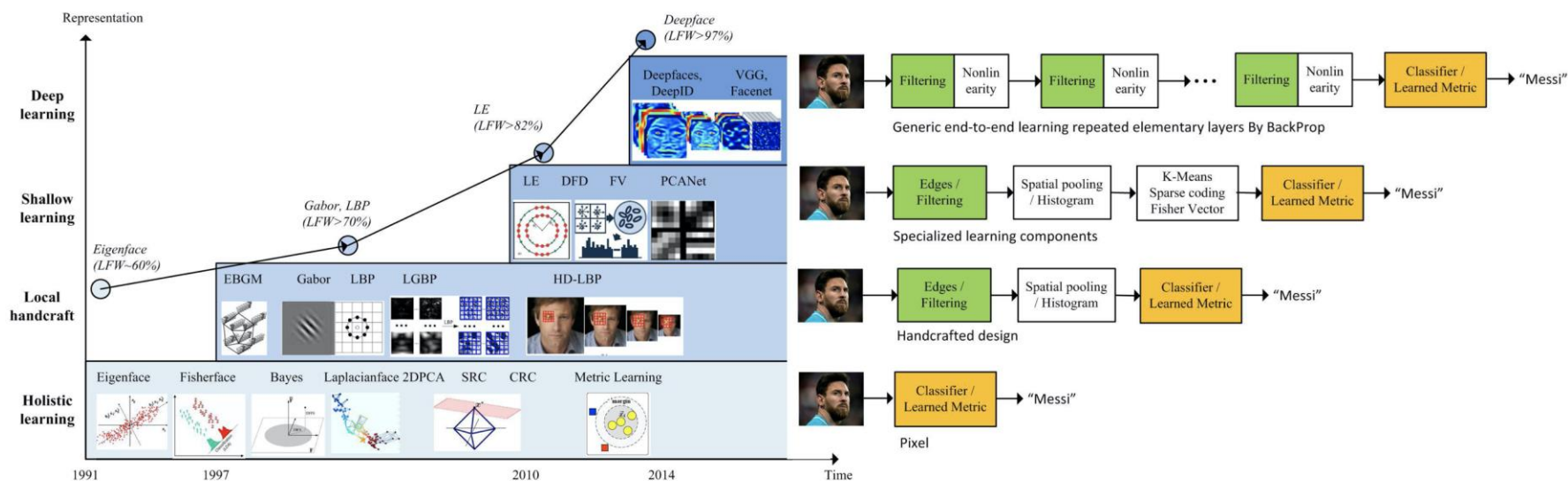
Lineární analýza prostoru obličejů

PCA

Local Binary Patterns a Histogramy

Metody založené na rysech

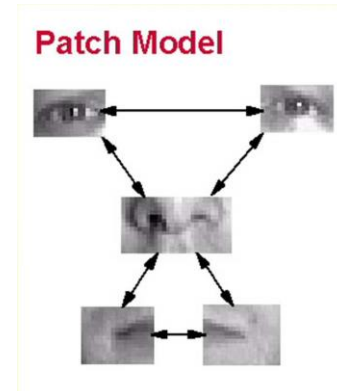
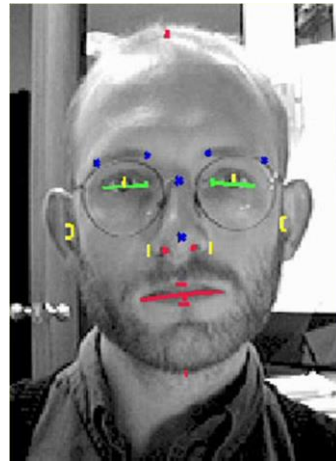
Holistické metody



Metriky obličeje

V obličeji se hledají (po kroku detekce) hlavní rysy, jako poloha nosu, úst a očí. Potom přichází detekce obočí, uší, rtů apod.

Porovnání probíhá na základě vzdáleností mezi jednotlivými body, přičemž se opět využívají toleranční limity.

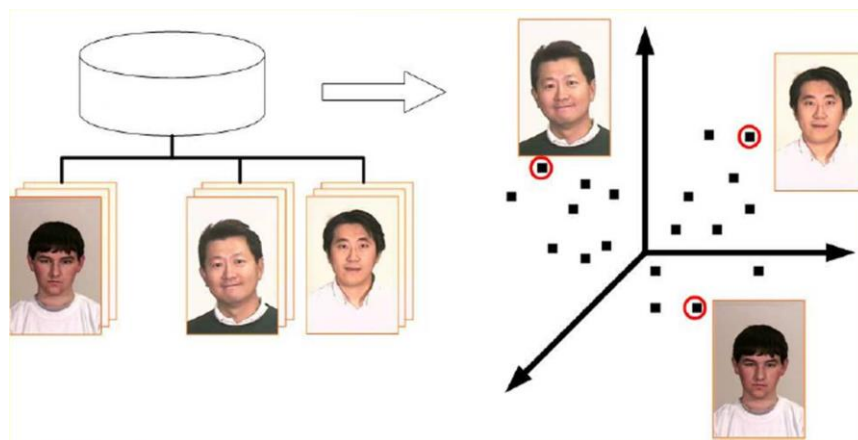


Vektorová reprezentace obrázku

2D obrázek je považován za vektor, provedeme-li spojení jednotlivých řádků (sloupců) do jednoho vektoru.

Obrázek obličeje odpovídá bodu v n -rozměrném prostoru R^n ($n = p \times q$, kde p je výška a q je šířka obrázku)

Prostor obličejů – množina obličejů rozvrstvených v prostoru R^n (distribuce obličejů v tomto prostoru).



Lineární analýza prostoru obličejů

Původní prostor obličejů o rozměrech $p \times q$ je příliš velký; data obličejů leží ve skutečnosti v prostoru mnohem nižší rozmanitosti.

Identifikace a parametrizace redukováného prostoru obličejů se zabývá lineární (podprostorová) analýza.

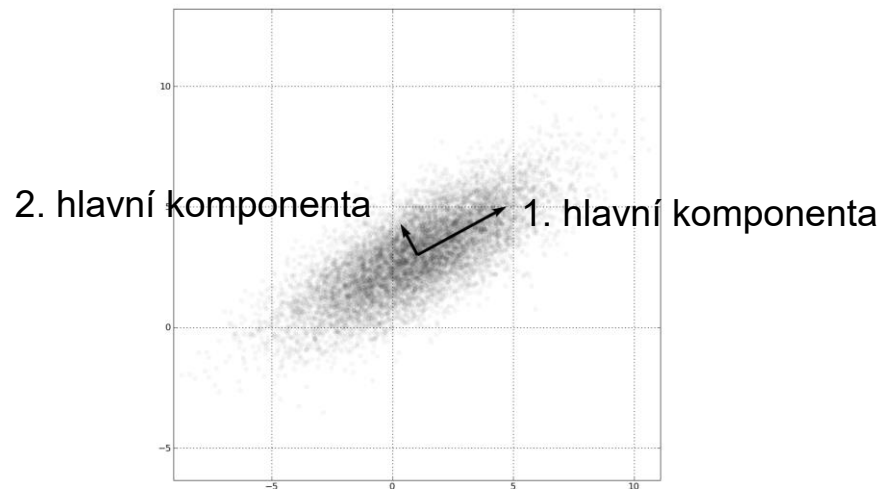
Lineární analýza – hledání lineární transformace $\mathbf{W}$, která mapuje originální vektor obličeje $\mathbf{x}$ na vektor projekčních koeficientů $\mathbf{y}$, tedy: $\mathbf{y} = \mathbf{W}^T \mathbf{x}$. Je-li d dimenzí $\mathbf{y}$ a n dimenzí $\mathbf{X}$, pak platí: $d \ll n$. Jedná se tedy o srovnávání vektorů příznaků a následné přiřazení do prostoru obličejů.

PCA - Principal Component Analysis

Analýza hlavních komponent kovarianční matice prostoru obličejů.

Hledáme projekci, kde jednotlivé dílčí projekční komponenty nejlépe vystihují distribuci dat.

Učení bez učitele.

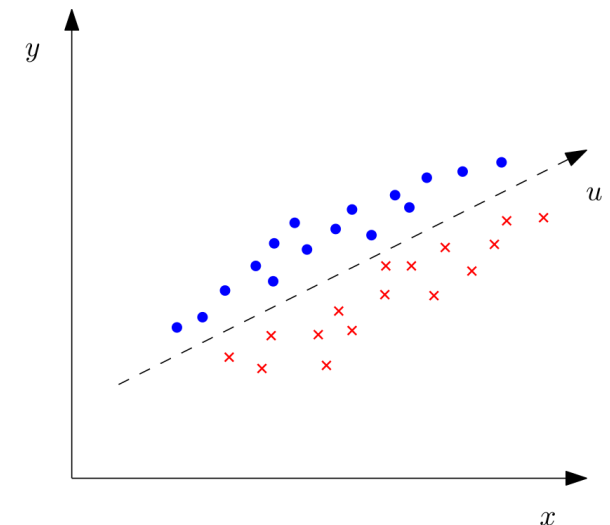
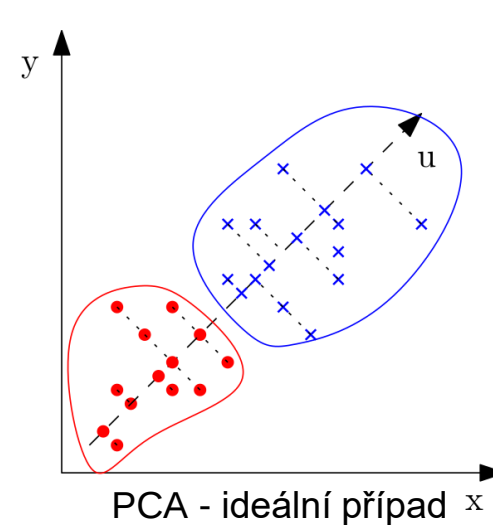


Výpočet kovarianční matice (matice rozptylu):

μ - průměrný vektor

$\mathbf{A} = [\mathbf{x}_1 - \mu, \mathbf{x}_1 - \mu, \dots, \mathbf{x}_N - \mu]$

$\mathbf{C} = \mathbf{A}\mathbf{A}^T$



PCA - Principal Component Analysis

Matice $A = N^2 \times M$, kde N je šířka obrázku (předpokládejme, že šířka i výška je stejná), M je množství vstupních vzorků.

Algoritmus učení

1. Kolekce $\mathbf{x}_i$ z n -dimenzionální množiny dat $\mathbf{X}$, $i=1,\dots,M$.
2. Centrování (korekce) všech bodů; spočti střední hodnotu μ a odečti od každého bodu: $\mathbf{x}_i \leftarrow \mathbf{x}_i - \mu$.
3. Spočti kovarianční matici $\mathbf{C}$ ($\mathbf{C} = \mathbf{A}\mathbf{A}^T$). Spočti kovarianční matici $\mathbf{C}$ ($\mathbf{C} = \mathbf{A}^T\mathbf{A}$).
4. Urči vlastní hodnoty a vlastní vektory matice $\mathbf{C}$.
5. Seřaď vlastní hodnoty (a odpovídající vlastní vektory) od největšího po nejmenší.
6. Vyber prvních k vlastních vektorů, ty tvoří projekční matici $\mathbf{W}$. Proved' prostřednictvím $\mathbf{C}$ projekci vlastních vektorů do původního prostoru.

Vlastní číslo λ a vlastní vektor $\mathbf{v}$ matice $\mathbf{A}$ splňují rovnici:

$$\mathbf{A} \mathbf{v} = \lambda \mathbf{v}$$

Eigenfaces (Eigenfaces, Turk 1991) – využití algoritmu PCA pro rozpoznávání obličeje

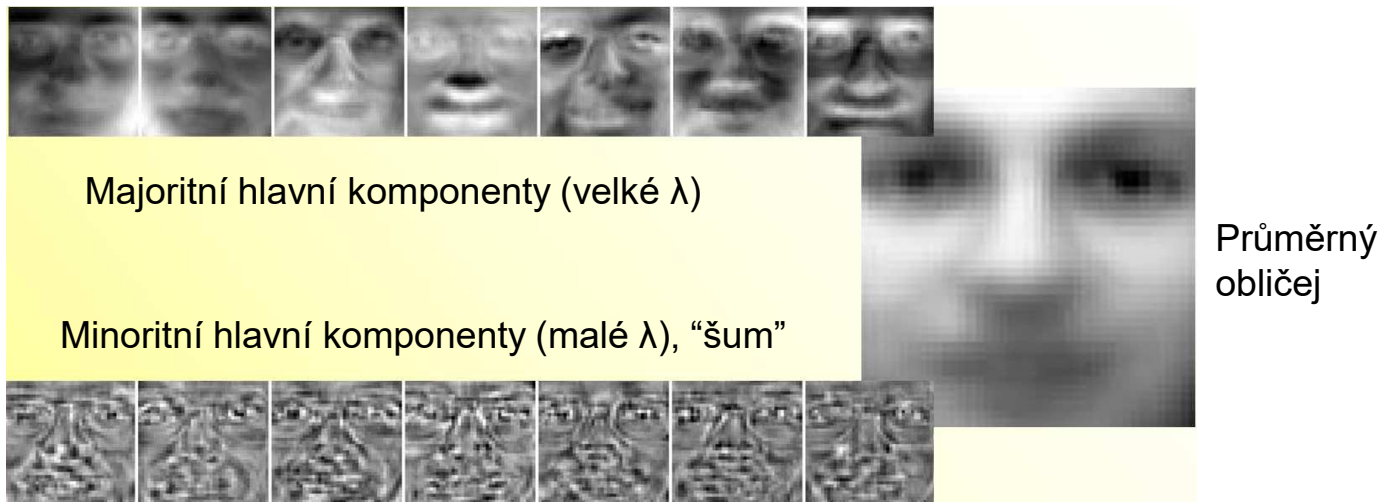
1. Množina snímků X , $i=1,\dots,N$, kde $\mathbf{x}_i$ je vektorová reprezentace snímku rozvinutého z matice $p \times q$ na vektor o velikosti $(p \times q) \times 1$.
2. Centrování (korekce) všech vektorů $\mathbf{x}_i$; spočti střední hodnotu μ a odečti od každého vektoru střední hodnotu: $\mathbf{x}_i \leftarrow \mathbf{x}_i - \mu$.
3. Spočti kovariační matici $\mathbf{C}$. Pokud ovšem vezmeme vzorec $\mathbf{C} = \mathbf{A}\mathbf{A}^T$, tak výsledná velikost bude $(p \times q) \times (p \times q)$. Zatímco pokud použijeme vzorec $\mathbf{C} = \mathbf{A}^T\mathbf{A}$, tak výsledná velikost matice $\mathbf{C}$ bude $(N) \times (N)$ -> v případě, že $N \ll (p \times q)$ je lepší zvolit právě tuto variantu. Ve druhém případě pak budeme pracovat s N vlastními vektory (*eigenvectors*) o velikosti N . Kovariační matici pro původní prostor budeme označovat $\mathbf{C}' = \mathbf{A}\mathbf{A}^T$.
4. Urči vlastní hodnoty a vlastní vektory matice $\mathbf{C}$. Ovšem je zapotřebí pamatovat na to, že vlastní vektory ($\mathbf{v}_i$) je zapotřebí přemapovat do původního prostoru, ke kterému se vztahuje matice $\mathbf{C}'$ a kde máme vlastní vektory označené jako $\mathbf{u}_i$. Pak přemapování provedeme pomocí $\mathbf{u}_i = \mathbf{A}\mathbf{v}_i$.
5. Seřaď vlastní hodnoty (a odpovídající vlastní vektory) od největšího po nejmenší.
6. Vyber prvních k vlastních vektorů ($k < M$), ty tvoří projekční matici $\mathbf{W}$.

PCA - Principal Component Analysis

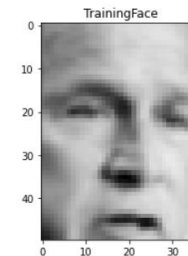
Projekční komponenty matice **W** připomínají obličeje. Libovolný obličej lze složit lineární kombinací projekčních komponent tzn. vynásobíme projekční matici váhovým vektorem. Následným složením projekčních komponent.

PCA - obecná statistická metoda pro analýzu dat

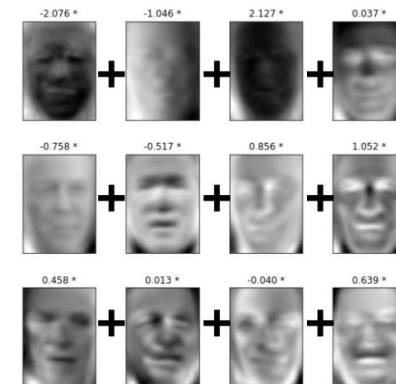
Eigenfaces - algoritmus pro rozpoznávání obličejů využívající PCA



$$x_i = \begin{bmatrix} w_1^i \\ w_2^i \\ w_3^i \\ \vdots \\ w_k^i \end{bmatrix}$$



=



Local Binary Patterns a Histogramy

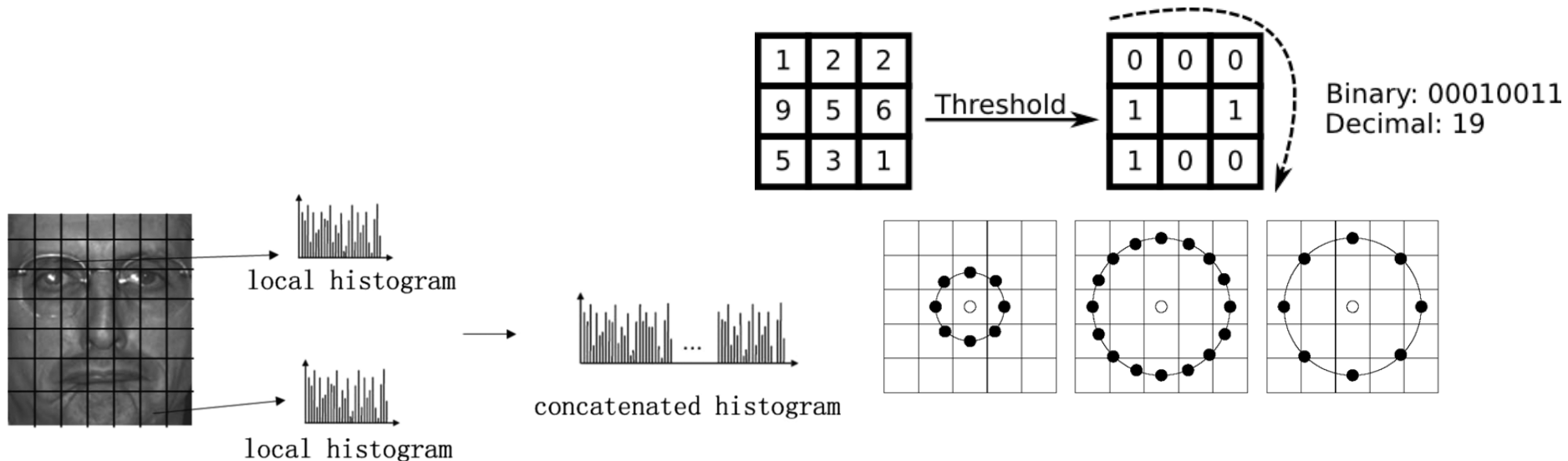
LBP (Ahonen 2006) se využívali hlavně pro klasifikaci textur.

Metoda nepotřebuje učení.

Každý bod obrázku je prahován se svým 8-okolím, ale existují i obecná rozšíření.

Obrázek se rozdělí do mřížky a v každé buňce se vypočítá histogram LBP.

Vektor rysů je tvořen spojenými histogramy.

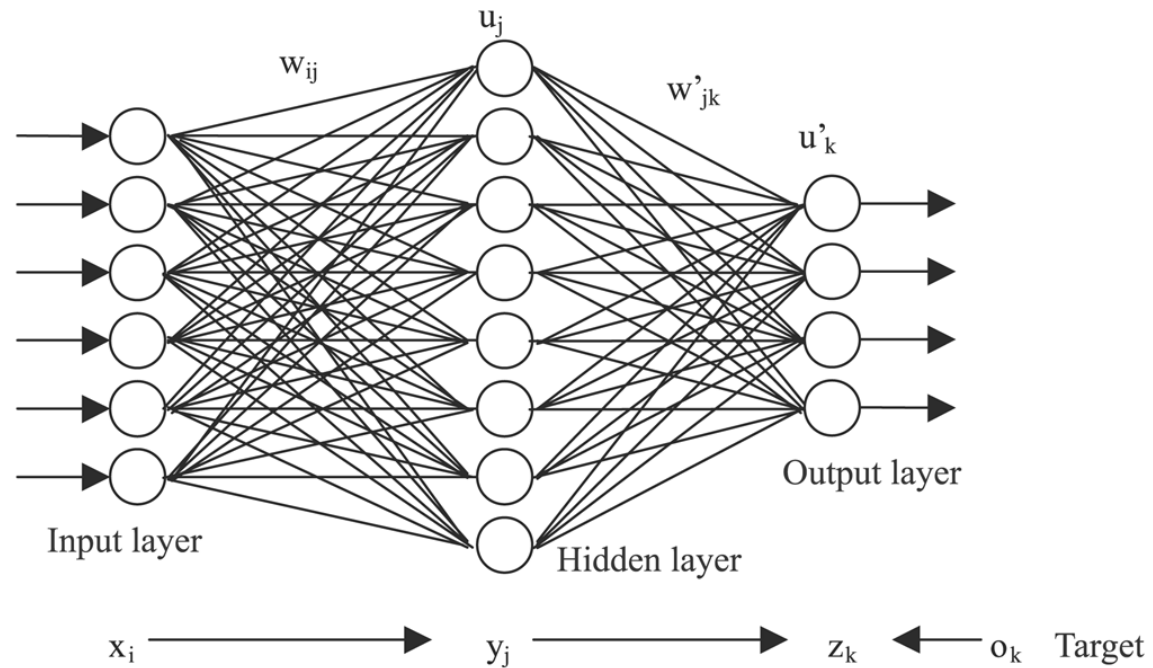


Neuronové sítě - klasifikace

N tříd

N výstupů

Výstup 1 z N



Jak natrénovat síť obecně? (Různé subjekty v trénovací a testovací DB)

Co dělat při zaregistrování nového uživatele?

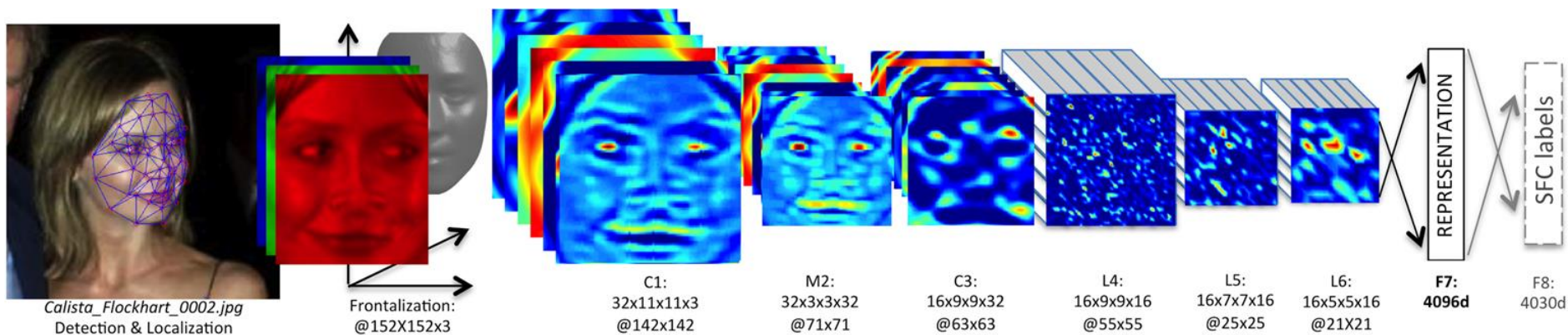
Přidat nový výstup a přetrénovat kompletně celou síť?

Hluboké neuronové sítě - úvod

Taigman 2014, Facebook

Síť obsahuje velkou spoustu vrstev a neuronů.

Je schopná se naučit, jaké filtry použít a jaké rysy extrahovat.



Pro učení se používá poslední vrstva (klasifikace 1:N).

Při reálném nasazení se používá jako vektor rysů výstup z předposlední vrstvy.

<http://deepdreamgenerator.com/>

<http://googleresearch.blogspot.co.uk/2015/06/inceptionism-going-deeper-into-neural.html>

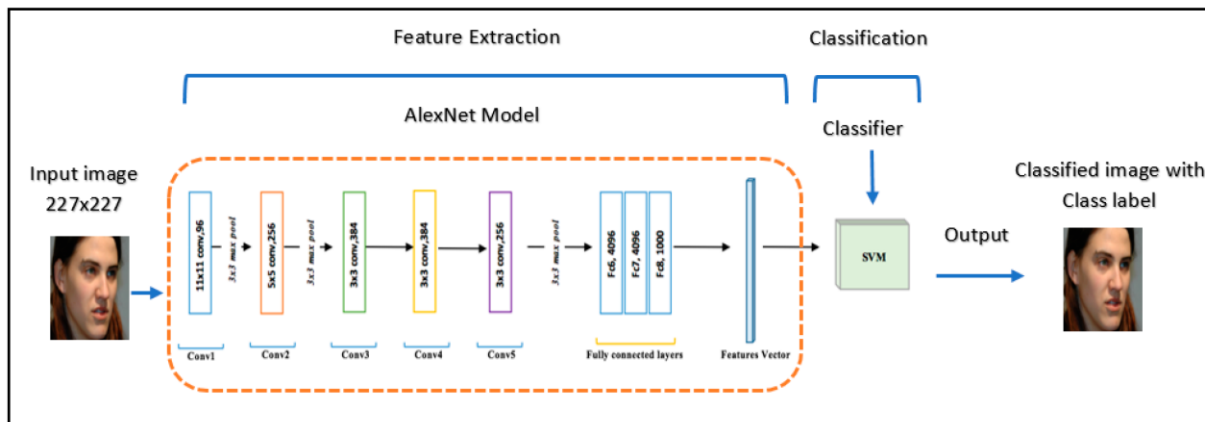
<https://facebook.com/download/233199633549733/deepface.pdf>

Neuronové sítě pro face embedding

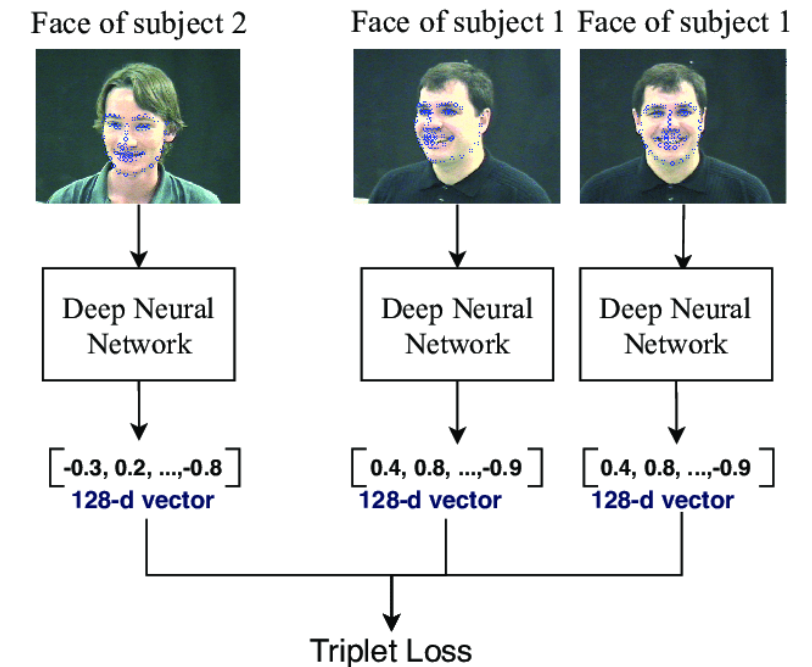
Face embedding - jedná se o proces převodu snímku obličeje na numerická data (vektor).

V dnešní době se k face embeddingu používají neuronové sítě, které jsou natrénované tak, aby maximalizovaly vzdálenosti výstupních vektorů mezi sebou pro co největší množinu vstupních dat rozdílných tříd.

Backbone – neuronová síť (obvykle konvoluční) pro extrakci příznaků.



<https://www.mdpi.com/2076-3417/9/20/4397>

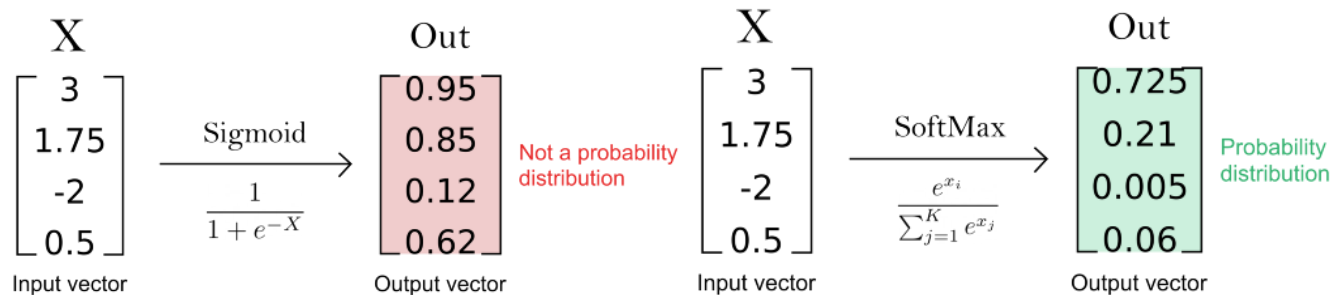


Neuronové sítě pro face embedding – ztrátové funkce

Vývoj neuronových sítí pro *face recognition* je v posledních letech zaměřený na **vylepšování ztrátových funkcí** a na **zlepšení možnosti zpracování velkých datasetů**.

Proč chceme upravovat ztrátové funkce? Chceme mít co nejblíže blíže extrahované vektory obličejů jedné třídy a zároveň je chceme mít, co nejdále od obličejů ostatních tříd.

Sigmoid vs Softmax



Neuronové sítě pro face embedding – ztrátové funkce

Softmax – jedna ze základních ztrátových funkcí pro *face embedding* ze které vycházejí další ztrátové funkce pro *face embedding*.

$$L = - \sum_{i=1}^m \ln \frac{\exp\{W_{y_i}^T \mathbf{x}_i + b_{y_i}\}}{\sum_{j=1}^n \exp\{W_j^T \mathbf{x}_i + b_j\}}$$

$\mathbf{x}_i$ – příznakový vektor pro i -tý obrázek

W_j – j -tý sloupec vah a b_j je příslušný bias

Center Loss Softmax (2016) – rozšíření ztrátové funkce Softmax o penalizaci pro body mimo těžiště (*centroid*) jejich třídy.

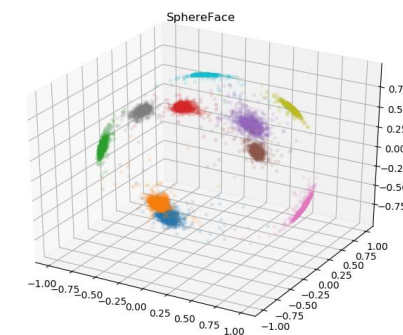
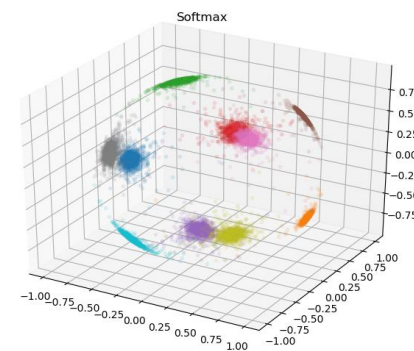
$$L = - \sum_{i=1}^m \ln \frac{\exp\{W_{y_i}^T \mathbf{x}_i + b_{y_i}\}}{\sum_{j=1}^n \exp\{W_j^T \mathbf{x}_i + b_j\}} + \frac{\lambda}{2} \sum_{i=1}^m \|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2$$

$\mathbf{c}_{y_i}$ – centroid třídy y_i pro vzorek s indexem i

A-Softmax (SphereFace) (2018) – ztrátová funkce založená na poznatku, že příznakové vektory jednotlivých tříd se v prostoru shlukují v podobných úhlech.

$$L = - \sum_i \ln \frac{\exp\left\{\|W_{y_i}\| \|\mathbf{x}_i\| \cos(\theta_{y_i,i}) + b_{y_i}\right\}}{\sum_j \exp\left\{\|W_j\| \|\mathbf{x}_i\| \cos(\theta_{j,i}) + b_j\right\}}$$

$\cos(\theta_{y_i,i})$ – úhel mezi příznakovým vektorem a příslušným sloupcem ve váhové matici



<https://github.com/4uiiurz1/keras-arcface>

<https://neptune.ai/blog/how-to-choose-loss-function-for-face-recognition>

Neuronové sítě pro face embedding – ztrátové funkce

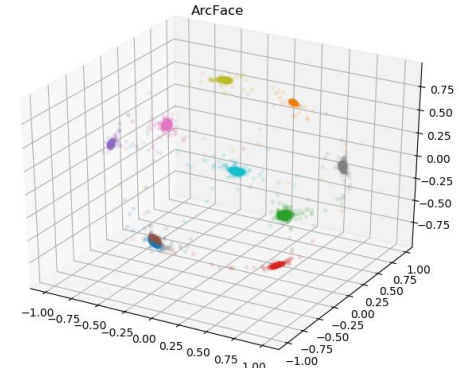
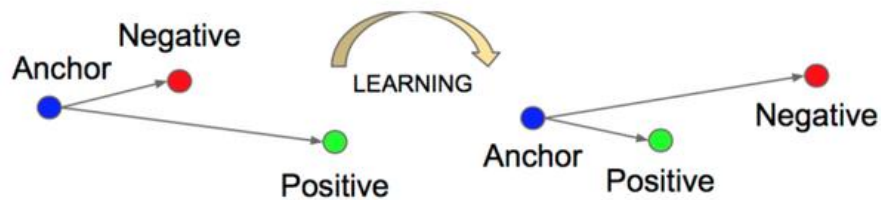
- Additive Angular Margin Loss (ArcFace – ~2018) – další ztrátová funkce, která využívá úhlové distribuce vektorů.

$$L = -\frac{1}{N} \sum_{i=1}^N \ln \frac{\exp \{s \cdot \cos(\theta_{y_i, i} + m)\}}{\exp \{s \cdot \cos(\theta_{y_i, i} + m)\} + \sum_{j \neq y_i} \exp \{s \cdot (\cos(\theta_{j, i}))\}}$$

θ_i – příznakový vektor pro i -tý obrázek

W_j – j -tý sloupec vah a b_j je bias

- Triple loss** (FaceNet) – ztrátová funkce, která je založená na vzdálenosti. Pracuje se třemi vstupy a to s *anchor*, s pozitivním vzorkem, který je stejné třídy jako *anchor* a negativním vzorkem, který je rozdílné třídy. Učením se snažíme maximalizovat vzdálenost negativního vzorku od *anchor* a naopak minimalizovat vzdálenost pozitivního vzorku od *anchor*. V případě *face recognition* je pozitivním vzorkem stejný obličej na jiném snímku. Negativní pak obličej odlišné osoby.



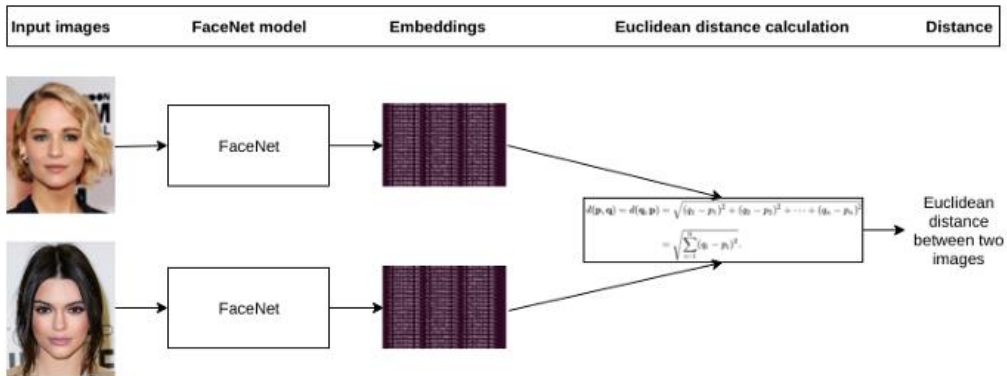
https://www.researchgate.net/figure/Embedding-vector-is-generated-using-a-deep-learning-model-involving-a-triplet-input-Two_fig3_351834316

<https://medium.com/analytics-vidhya/triplet-loss-b9da35be21b8>

Neuronové sítě pro face embedding

Metriky používané při porovnávání vektoru mezi sebou:

- Euklidovská vzdálenost
- Kosinova podobnost

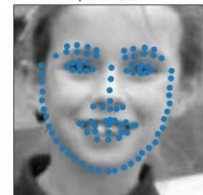
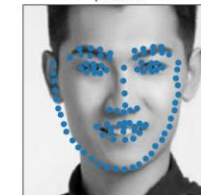
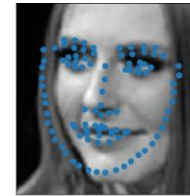
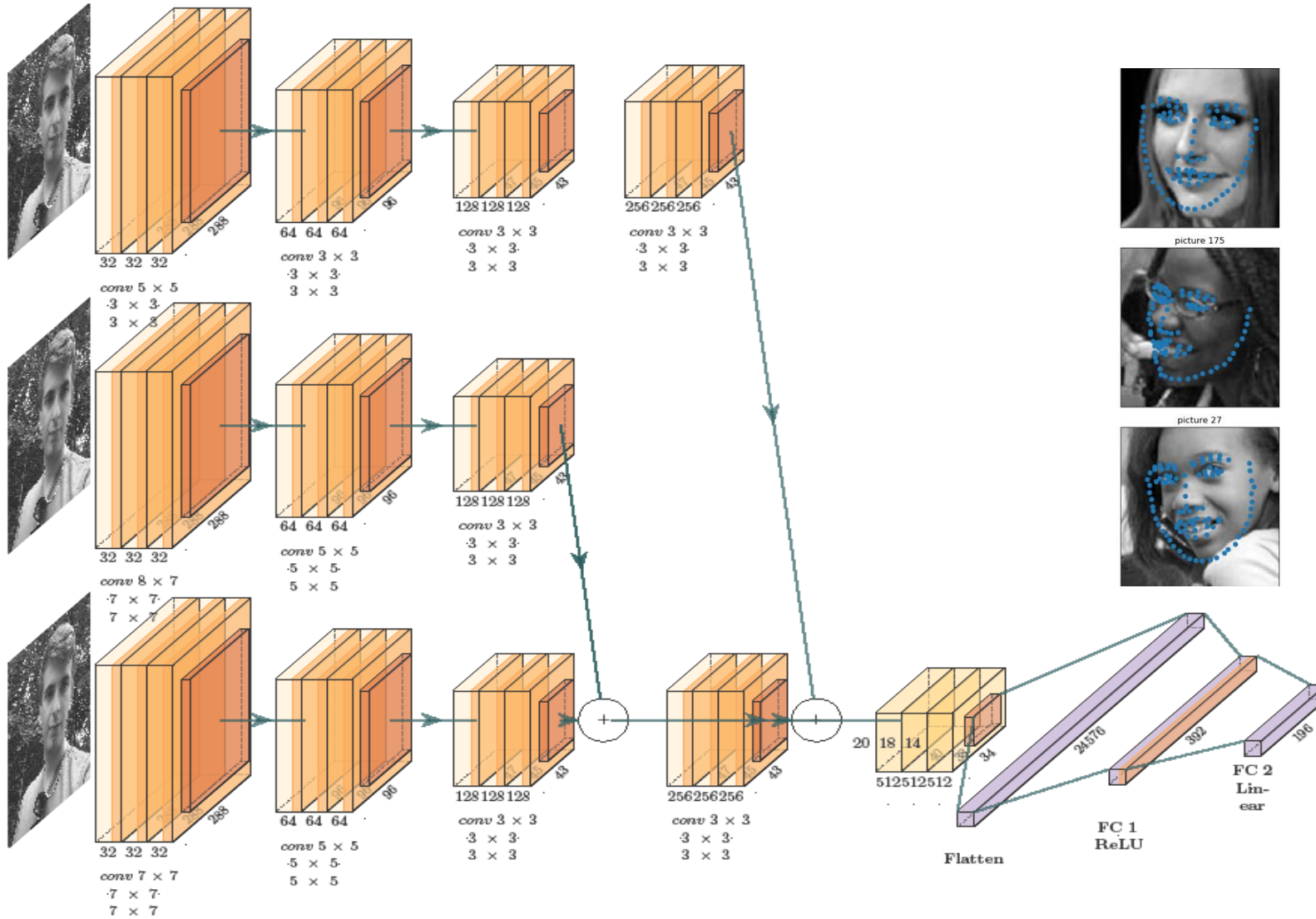


Příklady neuronových sítí pro face embedding

Framework deepFace (<https://github.com/serengil/deepface>) obsahuje následující algoritmy pro *face recognition*:

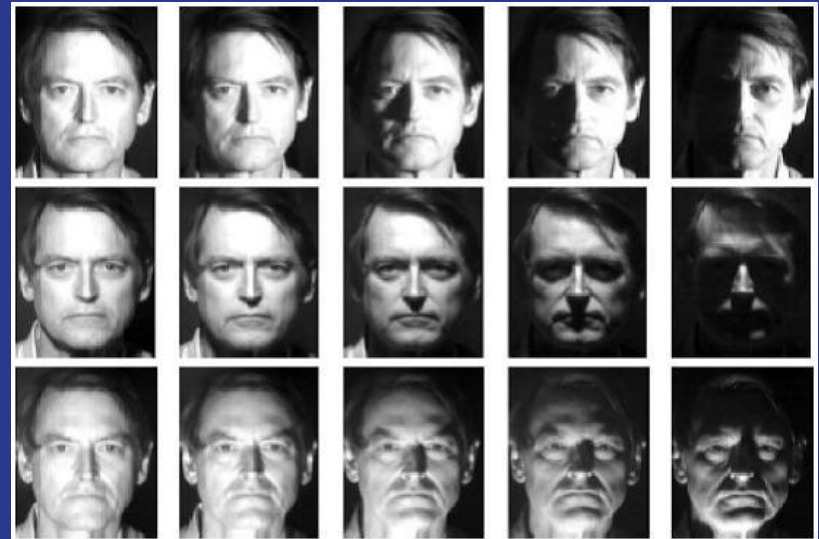
- VGG-Face (2015)
- Google FaceNet (2016)
- OpenFace (2016)
- Facebook DeepFace
- DeepID (2014)
- ArcFace (2018)

Sít' pro detekci významných bodů v obličeji



Rozpoznávání 2D obličeje

Datasey pro trénování a testování
neuronových sítí



Datasey pro trénování

Název datasetu	Počet identit	Počet snímků	Odkaz
CASIA-Webface	10k	0,5M	http://www.cbsr.ia.ac.cn/english/CASIA-WebFace-Database.html
CelebA	10k	0,2M	https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
UMDFace	8k	0,37M	http://umdfaces.io
MS1M-IBUG	85k	3,8M	
MS1M-ArcFace	85k	5,8M	
IMDB-Face	59k	1,7M	https://github.com/fwang91/IMDb-Face
MegaFace (train)	672k	4,7M	https://megaface.cs.washington.edu/dataset/download_training.html

CelebA – ukázka



Před využitím datasetu se seznamte se licenčními podmínkami. Ani to ovšem nemusí stačit....

Datasey pro testování

Název datasetu	Počet identit	Počet snímků	Odkaz
(LFW) - Labeled Faces in the Wild	5749	13233	http://vis-www.cs.umass.edu/lfw/
(CFP-FP) - Celebrities in Frontal-Profile in the Wild	500	0,7K	http://www.cfpw.io
AgeDB-30	570	12240	https://ibug.doc.ic.ac.uk/resources/agedb/
CALFW	5749	13233	http://whdeng.cn/CALFW/?reload=true
CPLFW	5749	13233	

LFW – ukázka



Přístupové systémy

Uživatelé zpravidla spolupracují - chtějí být rozpoznáni.

K dispozici je celá řada volně přístupných databází, které usnadní vývoj a testování systému:

FERET

FRGC

NLPR

BioID

<http://www.face-rec.org/databases/>



Komerční systémy

Innovatrics – IFace

Neurotechnology – MegaMatcher

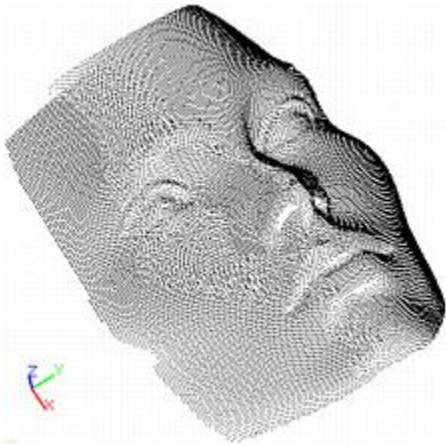
Idemia - MorphoKit

Quantasoft

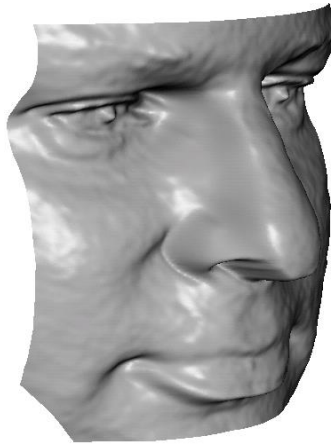
Rozpoznávání 3D obličeje



Reprezentace 3D modelu



Mračno bodů



Polygonální síť



Hlubková mapa
Parametrický povrch

Hlubková mapa: $(u,v) \rightarrow z$
Parametrický povrch: $(u,v) \rightarrow (x,y,z)$

3D Senzory - drahá komeční řešení



Artec 3D M

← Pouze 3D

3D s texturou →

Dodávaný SW umožňuje
fúzi dílčích snímků do
kvalitnějšího 3D modelu



Safran Morpho
Face Reader

Komerční
Biometrické
zařízení

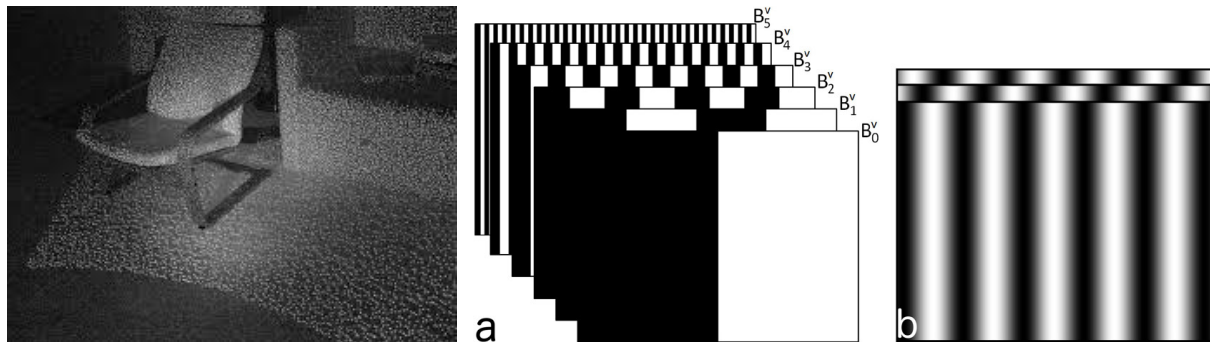
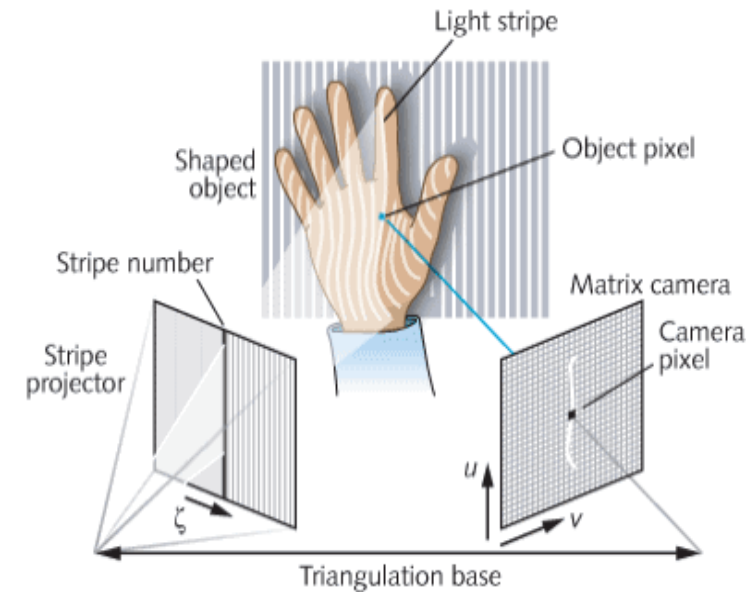


Minolta Vivid

Princip 3D snímání - strukturované světlo

1. Emitor vysílá (zpravidla v IR části spektra) specifický vzor.
2. Kamera to z jiného úhlu snímá.
3. Vypočítá se korespondence mezi body deformovaného vzoru a původním vzorem.
4. Ze znalosti vzájemné polohy emitoru a senzoru se triangulací vypočítají prostorové souřadnice

Podoba vzoru je často chráněna patentem. Může být v čase dynamická.



Princip 3D snímání - time-of-flight

Využívá konstantí rychlosti světla.

Emitor vysílá velmi krátké světelné pulzy.

Ty se na snímaném povrchu odrazí.

Vzdálenost snímaného objektu je přímo úměrná časovému rozílu mezi vysláním a přijetím odraženého pulsu.

Puls musí být velmi krátký (v řádu ns).

Nevýhodou je poměrně výrazný šum v z-ovém směru.



3D Senzory - velmi levné



Microsoft Kinect
1. gen. - strukturované světlo
2. gen. - Time of Flight



Occipital Structure.io
(structured light)



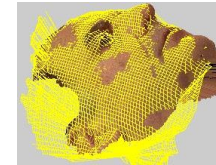
Softkinetic DepthSense
DS 325 (time-of-flight)



Intel RealSense
(structured light)

Algoritmy

Přímé porovnání povrchů



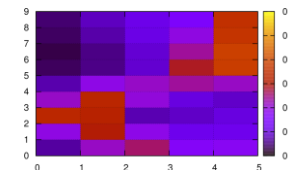
Adaptace 2D metod



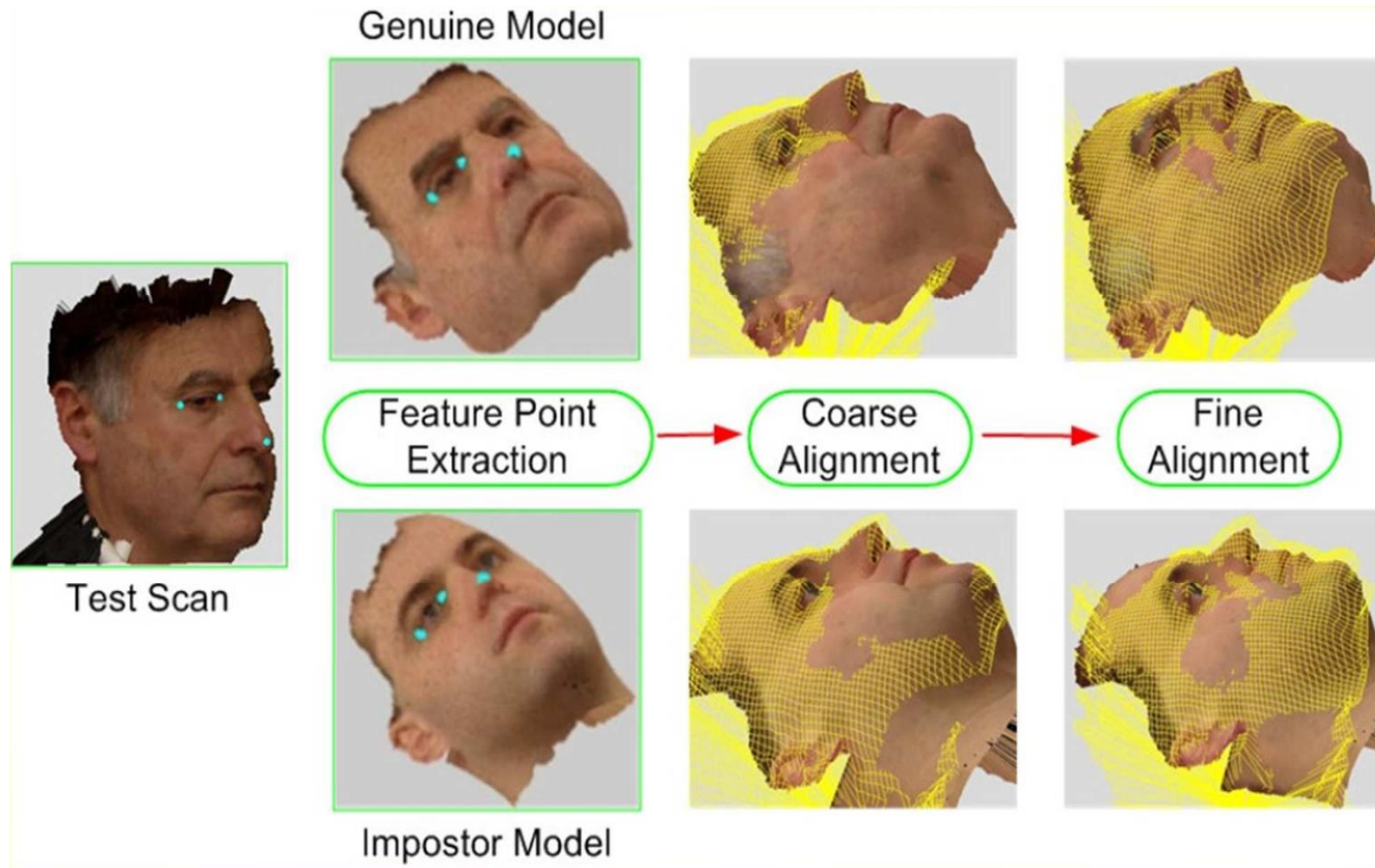
Isogeodetické křivky



Histogram z-ových souřadnic



Přímé porovnání povrchů



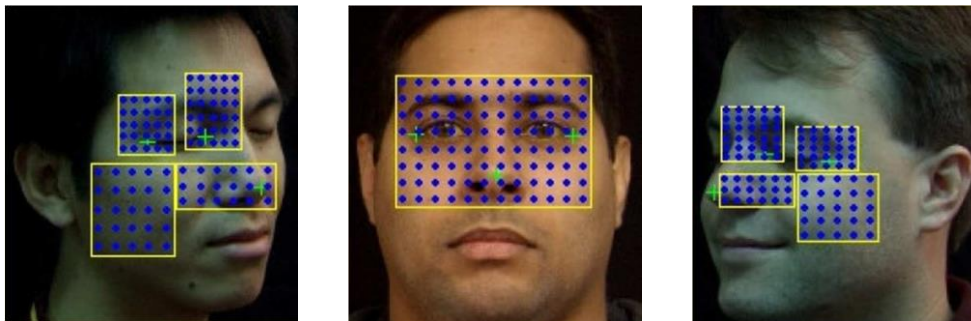
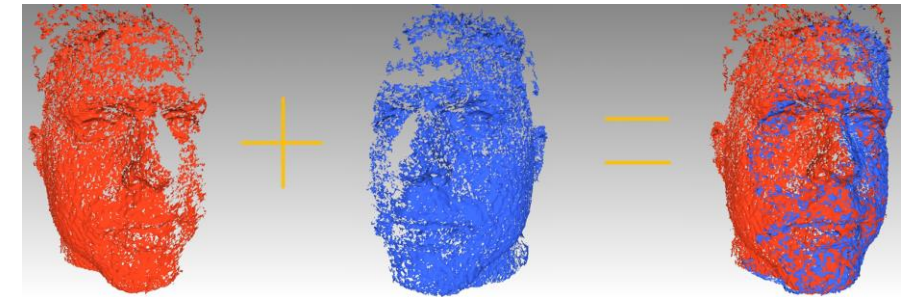
Přímé porovnání povrchů

Algoritmus:

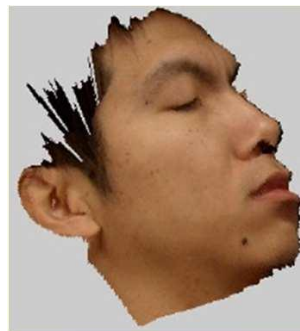
1. Vyhledání 3 bodů (kouty očí a špička nosu).
2. Hrubé zarovnání - nalezení optimální rigidní transformace pomocí korespondence 3 nalezených bodů.
3. Jemné zarovnání ICP algoritmem.
4. Výpočet podobnosti povrchů na základě součtu vzdáleností navzájem si odpovídajících bodů.

ICP algoritmus

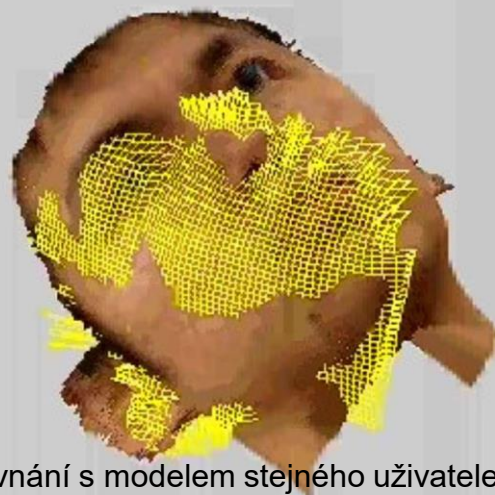
1. Vyber kontrolní body v 1. množině bodů.
2. Najdi nejbližší odpovídající body v 2. množině bodů.
3. Vypočítej optimální transformaci mezi oběma množinami.
4. Transformuj body.
5. Opakuj až do konvergence:
Výběr kontrolních bodů:
 - málo tvárné regiony
 - co největší pokrytí obličeje



Přímé porovnání povrchů - výsledky

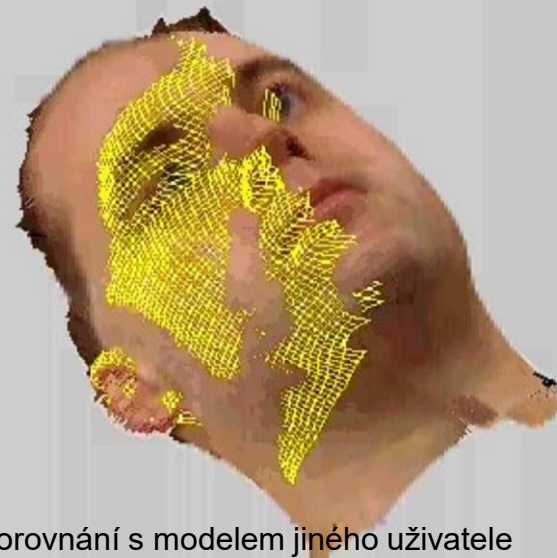


matching distance = 8.0444



Porovnání s modelem stejného uživatele

matching distance = 2.659



Porovnání s modelem jiného uživatele

Přímé porovnání povrchů - výsledky

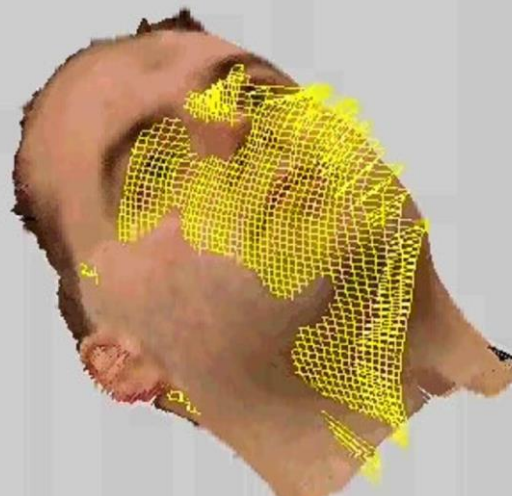


matching distance = 1,7864



Porovnání s modelem jiného uživatele

matching distance = 10.4618



Porovnání s modelem stejného uživatele

Adaptace 2D metod

Z 3D modelu můžeme získat různé reprezentace povrchu, textury nebo křivosti.

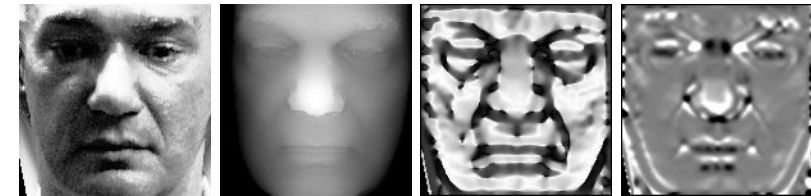
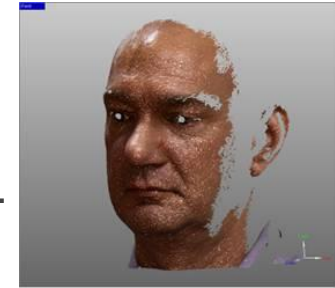
Takto získané obrázky lze dále zpracovávat 2D metodami pro rozpoznávání obličejů.

PCA

LDA

LBP

...



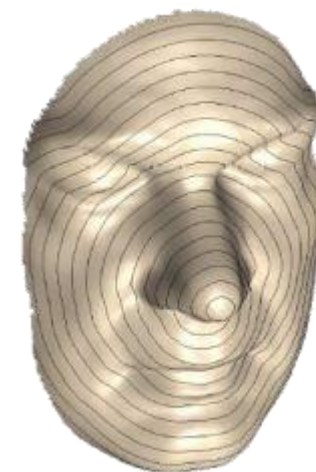
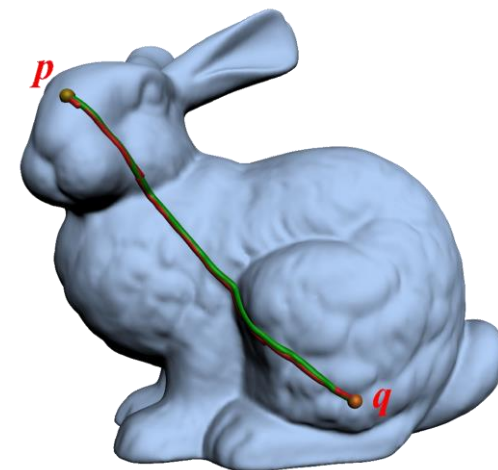
Isogeodetické křivky

Geodetická vzdálenost - nejkratší vzdálenost po povrchu mezi dvěma body.

Isogeodetická křivka - křivka spojující všechny body, které mají od daného středu stejnou geodetickou vzdálenost.

Pro výpočet isogeodesických křivek se používá tzv. **Fast Marching** algoritmus (varianta pro Dijkstrův algoritmus pro nalezení nejkratší cesty v grafu).

https://math.berkeley.edu/~sethian/2006/Explanations/fast_marching_explain.html



Isogeodetické křivky

Jahanbin 2008, http://live.ece.utexas.edu/publications/2008/sj_btas2008.pdf

Bronstein 2004, <http://www.cs.technion.ac.il/~ron/PAPERS/BroBroKimIJCV05.pdf>

Výhodou isogeodetických křivek je jejich invariance vůči rotaci obličeje - bez ohledu na to, jak je obličej natočený, je tvar křivky vždy stejný.

Rovněž jsou křivky relativně stabilní při různé mimice obličeje.

Z křivek lze extrahovat vektor rysů (VR):

Křivku navzorkovat a jednotlivé 3D souřadnice tvoří VR.

VR budou tvořit euklidovské vzdálenosti mezi středem křivek a dílčími navzorkovanými body.

Body z předchozích způsobů promítnou pomocí PCA.

...



Histogram z-ových souřadnic

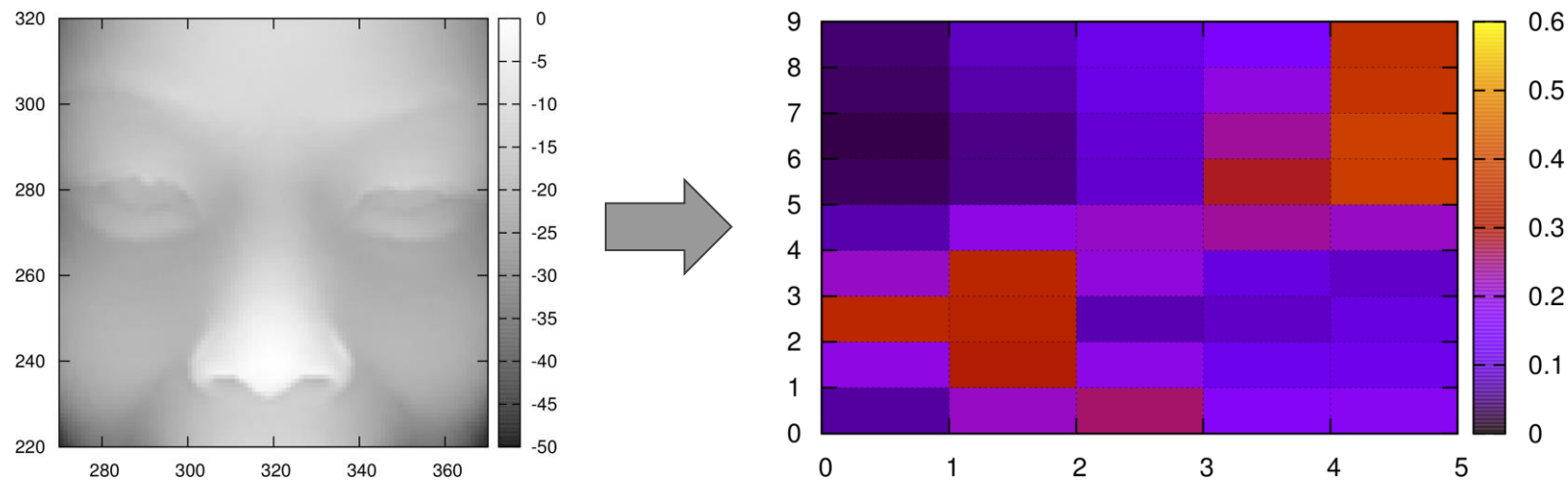
Zhou 2008, <http://www.christoph-busch.de/files/Zhou-HistogramFaceRec-EG08.pdf>

Vstupní 3D model převedeme na hloubkovou mapu nebo parametrický povrch.

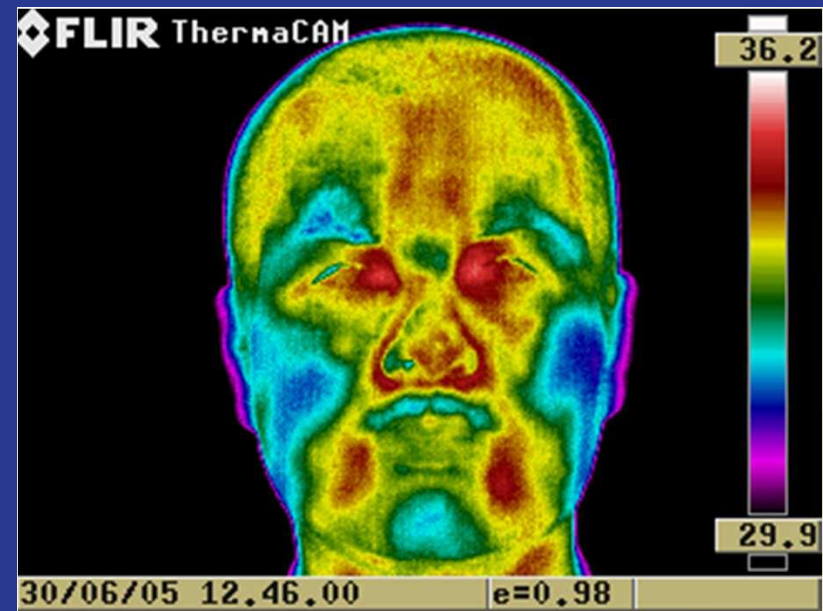
Mapu rozdělíme do N horizontálních pruhů.

V každém pruhu vypočítáme histogram z-ových souřadnic rozdělený do k košů.

Dostaneme vektor rysů o velikosti $N \times k$.



Termosnímký obličeje



Termosnímky obličeje

Pro záznam snímků se používá termokamera, u které je nutné správně nastavit (dodržet) následující parametry:

Emisní koeficient kůže: $e = 0,98$

Konstantní vzdálenost (správné zaostření)

Relativní vlhkost okolí

Teplota okolí

Záznamem jsou obrázky termomap obličeje (termogram).

Výhody: implicitní detekce živosti

Nevýhody: vyšší míra vnitrotřídní variability (tělesná teplota, emoční stav,...)



Termosnímký obličej

V termomapě obličej se hledají opět obdobným způsobem jako u klasického 2D rozpoznávání obličejů pozice očí, úst, nosu a hranice obličej.

Extrakce rysů může být provedena stejnými způsoby jako u 2D rozpoznávání



Ukázka vnitrotřídní variability