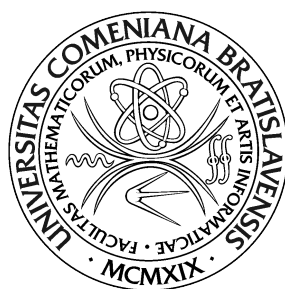


UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



VYHĽADÁVANIE KONKRÉTNEJ OSOBY VO VIDEU PODĽA VIZUÁLNEHO VZORU

Diplomová práca

UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



**Vyhľadávanie konkrétnej osoby vo videu podľa vizuálneho vzoru**

Diplomová práca

Študijný program: Aplikovaná informatika  
Študijný odbor: 2511 Aplikovaná informatika  
Školiace pracovisko: Katedra aplikovanej informatiky  
Školiteľ: RNDr. Zuzana Černeková, PhD.

Bratislava, 2018

Bc. Michal Hradečný



Univerzita Komenského v Bratislave  
Fakulta matematiky, fyziky a informatiky

## ZADANIE ZÁVEREČNEJ PRÁCE

**Meno a priezvisko študenta:** Bc. Michal Hradečný  
**Študijný program:** aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)  
**Študijný odbor:** aplikovaná informatika  
**Typ záverečnej práce:** diplomová  
**Jazyk záverečnej práce:** slovenský  
**Sekundárny jazyk:** anglický

**Názov:** Vyhľadávanie konkrétnej osoby vo videu podľa vizuálneho vzoru  
*Certain specific person search in video sequences based on visual example*

**Anotácia:** Naštudovať problematiku detekcie a sledovania ľudských tvárí. Analyzovať existujúce riešenia publikované v dostupnej odbornej literatúre. Vytvoriť databázu videozáznamov a hľadaných tvárí pre testovacie účely. Navrhnuť a implementovať metódu, ktorá vyhľadá osobu vo videozázname podľa zadaného vizuálneho vzoru t.j. tváre človeka. Vyhodnotiť dosiahnuté výsledky.

**Cieľ:** Naštudovať problematiku detekcie a sledovania ľudských tvárí. Analyzovať existujúce riešenia publikované v dostupnej odbornej literatúre. Vytvoriť databázu videozáznamov a hľadaných tvárí pre testovacie účely. Navrhnuť a implementovať metódu, ktorá vyhľadá osobu vo videozázname podľa zadaného vizuálneho vzoru t.j. tváre človeka. Vyhodnotiť dosiahnuté výsledky.

**Vedúci:** RNDr. Zuzana Černeková, PhD.  
**Katedra:** FMFI.KAI - Katedra aplikovanej informatiky  
**Vedúci katedry:** prof. Ing. Igor Farkaš, Dr.  
**Dátum zadania:** 24.10.2016

**Dátum schválenia:** 24.10.2016  
prof. RNDr. Roman Ďurikovič, PhD.  
garant študijného programu

.....  
študent

.....  
vedúci práce

Čestne prehlasujem, že som túto diplomovú prácu  
vypracoval/a samostatne s použitím citovaných zdrojov.

Bratislava, 2018

.....

Bc. Michal Hradečný

# Pod'akovanie

Chcem sa poďakovať svojej školiteľke RNDr. Zuzana Černeková, PhD. za cennú pomoc, rady, konzultácie a čas, ktorú mi venovala počas písania diplomovej práce.

# Abstrakt

Tento článok sa venuje problematike rozpoznávania ľudí na základe detegovanej tváre a natrénovaného klasifikátora s extrahovanými príznakmi. Popisuje proces rozpoznávania človeka porovnaním jeho tváre s trénovacou databázou tvárí. Tento proces zahŕňa najmä problém detekcie tváre a extrahovanie príznakov s najvyššou informačnou hodnotou. V článku je otestovaných viacero možných modelov pre obidva tieto problémy a porovnanie ich výsledkov. V prípade detekcie tvárí sú odtestované dva modely detekcie. Sú to modely Viola-Jones a DPM (Deformable parts model). Pre problém extrahovania príznakov práca obsahuje porovnanie rozpoznávania ľudí štyrmi rôznymi druhmi príznakov. Sú to príznaky HOG (Histogram of oriented gradients), LBP (Local Binary Pattern), Eigenface a Fisherface. Na záver je vyhodnotenie výsledkov týchto modelov, ich rýchlosti a schopnosti rozpoznávať ľudí podľa tváre v kombinácii s klasifikátormi SVM (Support Vector Machines) a KNN (K-nearest neighbors).

Kľúčové slová: rozpoznávanie, matlab, detekcia, klasifikátor, príznaky

# Abstract

This thesis is centered around the problem of recognizing people based on their detected face and trained classifier with extracted features. It describes the process of recognizing a person by comparing his face with a training database of faces. This process entails mostly the problem of face detection, coupled with extracting features with the highest information value. In this thesis we test multiple models for both of these problems and evaluate their results. In the case of face detection two models of detection were tested. These are Viola-Jones and DPM (Deformable parts model). For the problem of feature extraction this thesis contains the comparison of recognition of people using four different types of features. These are HOG (Histogram of oriented gradients), LBP (Local Binary Pattern), Eigenface and Fisherface features. The conclusion contains the evaluation of these models, their speed and ability to recognise people by their face in combination with classifiers SVM (Support Vector Machines) and KNN (K-nearest neighbors).

Keywords: recognition, matlab, detection, classifiers, features

# Obsah

<b>1</b>	<b>Cieľ práce</b>	<b>1</b>
<b>2</b>	<b>Motivácia</b>	<b>2</b>
<b>3</b>	<b>Úvod</b>	<b>3</b>
<b>4</b>	<b>Prehľad problematiky</b>	<b>5</b>
4.1	Proces rozpoznávania . . . . .	5
4.1.1	Model rozpoznávania tvárí . . . . .	6
4.1.2	Vytvorenie databázy . . . . .	7
4.2	Detekcia tváre . . . . .	8
4.2.1	Viola-Jones . . . . .	8
4.2.2	Deformable parts model (DPM) . . . . .	10
4.2.3	Neurónové siete . . . . .	12
4.3	Extrahovanie príznakov . . . . .	13
4.3.1	Typy príznakov do trénovacej množiny . . . . .	14
4.3.2	Local Binary Pattern (LBP) . . . . .	15
4.3.3	Histogram of oriented gradients (HOG) . . . . .	16
4.3.4	Eigenfaces . . . . .	18
4.3.5	Fisherfaces . . . . .	20
4.4	Klasifikácia príznakov . . . . .	23



<i>OBSAH</i>	ix
4.4.1 K-nearest neighbors (KNN) . . . . .	23
4.4.2 Support Vector Machines (SVM) . . . . .	24
<b>5 Návrh riešenia</b>	<b>26</b>
5.1 Návrh používateľského rozhrania (GUI) . . . . .	26
5.2 Postup algoritmu rozpoznávania . . . . .	27
<b>6 Implementácia</b>	<b>29</b>
6.1 Implementácia detegovania tvárí . . . . .	29
6.2 Implementácia extrahovania príznakov a rozpoznávania . . . . .	30
<b>7 Výsledky</b>	<b>33</b>
7.1 Meranie presnosti . . . . .	34
7.1.1 Testovacia databáza . . . . .	36
7.2 Výsledky detekcie tvárí . . . . .	37
7.2.1 Detekcia pomocou Viola-Jones . . . . .	37
7.2.2 Výsledky Deformable parts model (DPM) detekcie . . . . .	41
7.2.3 Porovnanie detektorov . . . . .	44
7.3 Výsledky rozpoznávania osôb . . . . .	45
7.3.1 Rozpoznávanie pomocou HOG príznakov . . . . .	45
7.3.2 Rozpoznávanie pomocou LBP príznakov . . . . .	47
7.3.3 Rozpoznávanie pomocou Eigenface príznakov . . . . .	48
7.3.4 Rozpoznávanie pomocou Fisherface príznakov . . . . .	49
7.3.5 Porovnanie výsledkov . . . . .	49
<b>8 Záver</b>	<b>52</b>

# Kapitola 1

## Cieľ práce

Naštudovať problematiku detekcie a sledovania ľudských tvárí. Analyzovať existujúce riešenia publikované v dostupnej odbornej literatúre. Vytvoriť databázu videozáznamov a hľadaných tvárí pre testovacie účely. Navrhnuť a implementovať metódu, ktorá vyhľadá osobu vo videozázname podľa zadaného vizuálneho vzoru t.j. tváre človeka. Vyhodnotiť dosiahnuté výsledky.

# Kapitola 2

## Motivácia

Mojou motiváciou k vypracovaniu témy rozpoznávania ľudí na základe vizuálneho vzoru je celkový záujem o grafické a vizualizačné témy v programátorskom svete. Taktiež ma láka fakt, že sa jedná o tému z reálneho života, keďže podobné systémy sa dajú využiť v kamerových systémoch na rôzne účely. Môže sa jednať o rozpoznávanie zločincov z kamerových záznamov, kontrolovanie dochádzky na pracovisku, či len overenie používateľa kamerou na smartfóne. Zaujímajú ma aj problémy, ktoré momentálne výskumníci v danej oblasti riešia, keďže podobné systémy na detegovanie a rozpoznávanie ľudí z fotiek ale najmä z videí sú stále pomerne málo rozšírené v bežnom živote. Problematika detegovania a rozpoznávania ľudí sa podľa naštudovaných zdrojov rozvíja minimálne od konca 90.rokov minulého storočia. Verím, že s pomocou školiteľa sa mi podarí porozumieť danej problematike a taktiež prispieť svojím vlastným výskumom.

# Kapitola 3

## Úvod

Rozpoznávanie tvári pomocou počítačových aplikácií je v dnešnej digitalizovanej dobe, kde na každom kroku na ľudí striehne na stenách kamera, čoraz aktuálnejšia téma. Typické využitie takýchto aplikácií je v bezpečnostných systémoch. Z digitálneho obrázka alebo videa sa dajú identifikovať alebo verifikovať osoby na základe porovnania črt ich tváre s dostupnou databázou ľudí a ich tvárí.

Takéto systémy majú veľký rozsah využitia. Môžeme sa s nimi bežne stretnúť pri cestovaní na letiskách alebo hraničných priechodoch, na porovnanie tvárí s fotografiami v ich cestovných pasoch alebo vodičských preukazoch, pri vchodoch do štadiónov, či dokonca na zabránenie volebných podvodov. Kým rozpoznávanie osôb má veľké využitie v bezpečnostných systémoch na identifikáciu potenciálnych zločincov či teroristov, jeho využitie nekončí len pri bezpečnosti. V poslednej dobe sa začali takéto systémy využívať aj v komerčnej sfére života. Napríklad v mobilných telefónoch na ich odomknutie pomocou prednej kamery, v digitálnych kamerách na zameranie a odmeranie osvetlenia tváre, či na kontrolu dochádzky v práci. Aplikácie na rozpoznávanie tvárí osôb majú však svoje problémy a nie sú vždy presné. Ob-

čas priradia nespravnu osobu z databazy k nasnimanej osobe a veľký problém spočíva taktiež v otočení, poprípade osvetlení tváre na snímkach.

Rozpoznávanie tváre (face recognition) je podtrieda rozsiahlejšieho odboru rozpoznávania objektov (object recognition) a v mnohých prípadoch používa rovnaké metódy výpočtov.

# Kapitola 4

## Prehľad problematiky

### 4.1 Proces rozpoznávania

Rozpoznávanie tvárí je zložitý proces, ktorý sa skladá z viacerých podprocesov. Na dosiahnutie efektívneho systému rozpoznávania treba všetky tieto podprocesy riešiť. Jednotlivé procesy sa dajú vytriediť do blokov a vytvoriť model, ktorý je znázornený na obr. 4.1.



Obr. 4.1: Model rozpoznávania tvárí [Ban14]

### 4.1.1 Model rozpoznávania tvári

Opis blokov modelu rozpoznávania tvári [Ban14]:

- Vstup - Vstupný obrázok s tvárou osoby na identifikáciu alebo verifikáciu.
- Predspracovanie – využíva sa, ak je potrebné vstupné dáta špeciálne upraviť pre rozpoznávací systém, napr. detekcia tváre v obraze, odstránenie šumu alebo vrátenie vstupných dát a žiadosť o opätovné zosnímanie. Vstupom je zosnímaná tvár a výstupom je upravený obraz.
- Normalizácia – Normalizácia tváre, spočíva napríklad v úprave pozície a natočenia tváre alebo zmene v škálovaní.
- Extrakcia príznakov – extrahuje charakteristiky s najvyššou informačnou hodnotou; výstupom je vektor príznakov. Pri trénovaní systému sa tieto vektory ukladajú do databázy, pri rozpoznávaní postupuje vektor do ďalšieho - porovnávacieho bloku.
- Porovnávací blok – porovnáva vstupný vektor príznakov s vektormi uloženými v databáze, na základe určenej klasifikačnej metódy.
- Rozhodovací blok – na základe predošlých výpočtov určí (alebo overí) identitu vstupnej osoby, prípadne zamietne vstupnú osobu, ak sa v danom systéme nenachádza.
- Výstup – výstupné zariadenie, napr. displej, zobrazí výsledok rozpoznávania - identita alebo záznam vstupnej osoby, prípadne zamietnutie.

### 4.1.2 Vytvorenie databázy

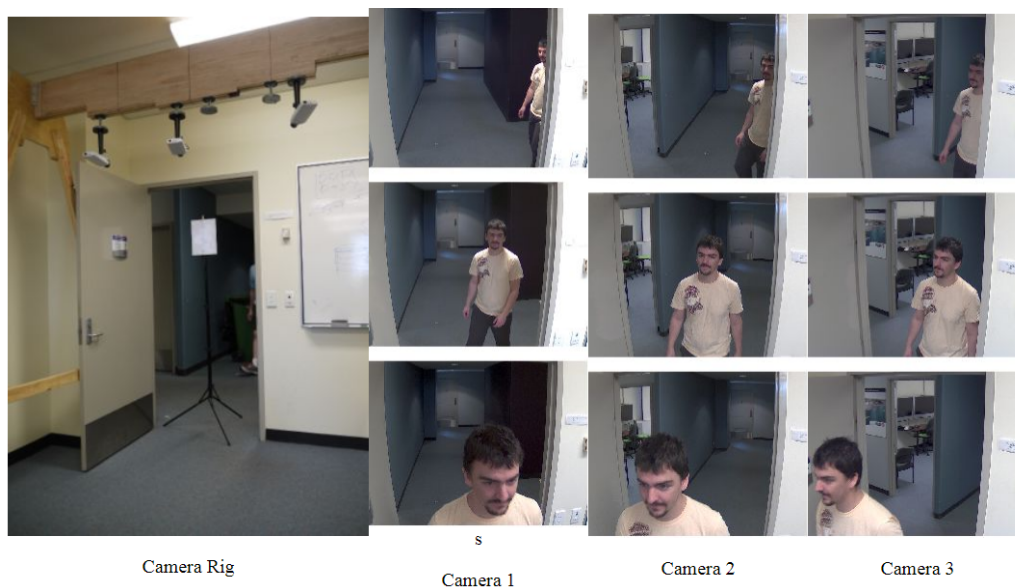
Na otestovanie efektívnosti vytvorenej aplikácie na vyhľadávanie konkrétnej osoby na videu je potrebné mať k dispozícii dostatočne veľkú testovaciu databázu. Databáza by mala obsahovať obrázky tvári osôb nafotených spredu, ako aj z rôznych uhlov. Taktiež je treba mať k dispozícii dostatočne veľkú vzorku videí, v ktorých sa nachádzajú nafotené osoby z databázy. Keďže jedným z cieľov práce je taktiež vytvoriť testovaciu databázu, je potrebné vybrať si vhodnú množinu objektov, ku ktorým sa dajú nájsť ich profily tváre a taktiež videá. Jednou takouto možnosťou je databáza obsahujúca celebrity, ku ktorým je dostupné veľké množstvo fotiek a videí (interview, úryvky filmu, ukážky) z internetu.

Nevýhodou vlastnej databázy je chýbajúca možnosť porovnania výsledkov s inými podobnými prácami na štandardných databázach, ako napr. AT&T databáza tvári.

Druhou možnosťou je použiť voľne dostupné databázy z internetu. V našom prípade je pri výbere databázy nevyhnutnou podmienkou obsah videí s postavami na verifikáciu a taktiež tváre ľudí vyskytujúcich sa na daných videách v rôznych uhloch. Typická databáza na účely testovania algoritmov detegovania a rozpoznávania tvári obsahuje jednotlivé fotky ľudí v odlišných prostrediach a ich tváre. Databázy videozáznamov a tvári sú pomerne vzácne. Jedna takáto voľne dostupná video databáza sa nazýva ChokePoint databáza [WCM<sup>+</sup>11]. Obsahuje videozáznamy z chodieb pracoviska z 3 rôznych uhlov v dvoch lokalitách. Oproti videám celebrit z internetu táto databáza simuluje reálne kamerové systémy na pracoviskách.

Pre zaujímavosť sa dajú vyskúšať algoritmy detegovania a rozpoznávania tvári aj na kamerových záznamoch z knižnice v informačnom pavilóne FMFI. Ukážka Chokepoint databázy je na obr. 4.2.



Obr. 4.2: Kamerový systém ChokePoint databázy [WCM<sup>+</sup>11]

## 4.2 Detekcia tváre

Predtým, než budeme môcť rozhodnúť či sa na obrázku nachádza niektorá osoba z databázy, musíme byť najprv schopný nájsť všetky tváre na obrázku, ktoré chceme porovnávať (poprípade určiť či sa na obrázku vôbec nejaká tvár nachádza). Na tento účel existuje viacero možných algoritmov.

### 4.2.1 Viola-Jones

Jedným z najpoužívanějších je algoritmus Viola-Jones [VJ01], ktorý slúži primárne na detegovanie frontálnych tvári na obrázku, ale dá sa natrénovať na detegovanie rôznych objektov (napr. hľadanie horných častí tela, nosov, očí). Viola-Jones využíva podobnosť ľudských tvári na ich detegovanie [BP16]. Napríklad okolie očí býva tmavšie ako líčne kosti, alebo nos býva svetlejší ako okolie očí. Tieto príznaky ľudskej tváre sú reprezentované v takzvaných

Haarovych príznakov, znázornené na obr. 4.3 a obr. 4.4. Hodnota príznaku v Haarovych príznakov je vypočítaná ako suma pixlov v svetlej časti odpočítaná od sumy pixlov v tmavej časti štvoruholníka. Ukážka detekcie pomocou Viola-Jones algoritmu je na obr. 4.5.



Obr. 4.3: Haarov príznak, ktorý sa podobá okoliu očí aplikovaný na obrázok



Obr. 4.4: Haarov príznak, ktorý sa podobá nosu aplikovaný na obrázok

Nevýhodou algoritmu Viola-Jones je, že kvôli jednoduchosti Haarovych príznakov je nespoľahlivý v nekontrolovaných podmienkach, kde tvár je v rôznych pózach a natočeniach, alebo je nezvyčajne osvetlená. Tento detektor poskytuje vysokú rýchlosť detekcie (často používaný v real-time systémoch detekcie), ale iba priemernú presnosť detekcie.



Obr. 4.5: Ukážka Viola-Jones detekcie

### 4.2.2 Deformable parts model (DPM)

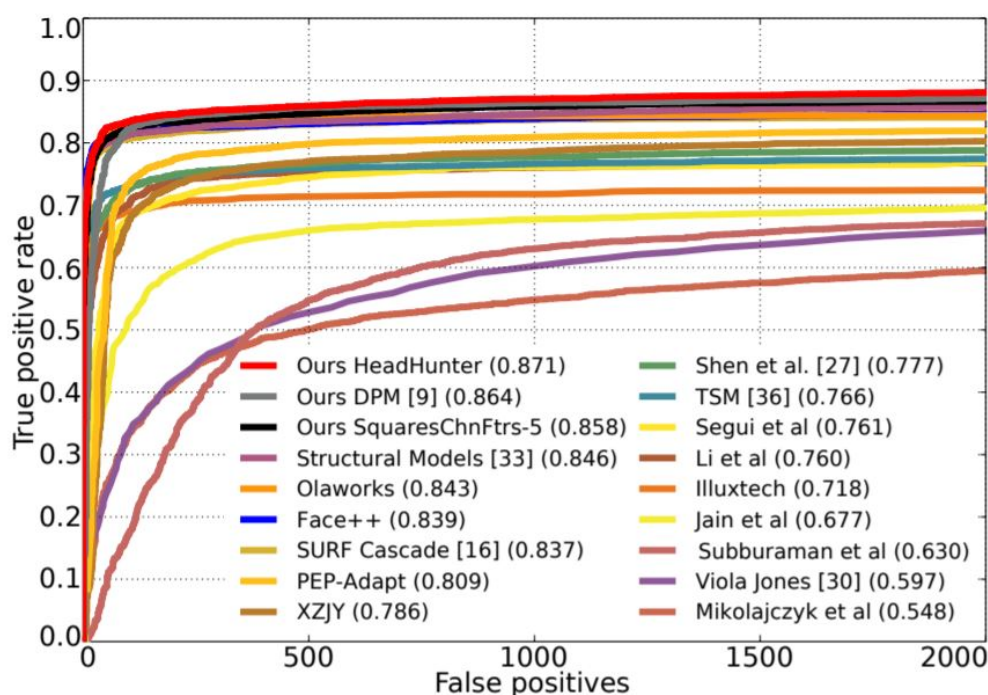
Vďaka jeho silným výsledkom detekcie sa DPM model presadil ako de-facto štandard generickej detekcie objektov [MBPVG14]. DPM predpokladá, že objekt sa skladá z viacerých jeho častí. Detektor najprv nájde zhodu s celým objektom a potom používa model jeho častí na upravenie výsledku, ako znázornené na Obr. ?? [ZR12].

Vďaka princípu deformovateľnosti jednotlivých častí objektu vie DPM model identifikovať aj čiastočne zakryté alebo zmenené objekty. Obr. 4.6 znázorňuje model deformovateľných častí ľudskej tváre, ktorý nemá problém identifikovať tvár ani s rôznymi výrazmi tváre a grimasami.



Obr. 4.6: DPM model tváre v rôznych natočeniach s HOG príznakmi [ZR12]

DPM detektor nie je známy pre svoju rýchlosť. Jeho prednosťou však je schopnosť identifikovať zložité objekty. Na obr. 4.7 je vidieť porovnanie efektívnosti viacerých známych detektorov. Správne natrénovaný DPM detektor sa umiestnil na druhom mieste v danom teste detektorov, naproti tomu Viola-Jones algoritmus dosiahol druhý najhorší výsledok. Tento test však neberie do úvahy rýchlosť jednotlivých detektorov a využiteľnosť v reálnych podmienkach.



Obr. 4.7: Porovnanie detektorov tvárí [MBPVG14]

### 4.2.3 Neurónové siete

Ďalšia možnosť nájdania tváre v obrázku zahŕňa využitie konvolučných neurónových sietí (Convolutional Neural Network - CNN) [TYRW14]. Tieto neurónové siete treba natrénovať tréningovou množinou negatívnych tréningových prvkov a množinou tvárí osôb v rôznych pózach a natočeniach. Platí, že čím väčšia je tréningová množina neurónových sietí, tým väčšia je ich presnosť. To je aj ich nevýhoda, keďže na vytvorenie efektívnej neurónovej siete je potrebné mať veľké množstvo (niekoľko tisíc) prvkov v tréningovej množine. Z tohto dôvodu sme sa neurónovými sieťami podrobnejšie nezaoberali. Vďaka širokým možnostiam natrénovania neurónovej siete, je možné lepšie detegovať aj tváre vo väčších natočeniach a s rôznymi grimasami na tvári. Obidva

spomenuté spôsoby detegovania tváre využívajú takzvanú kaskádovú architektúru (Cascade architecture). Úlohou tejto architektúry je znížiť výpočtovú náročnosť daných algoritmov, využitím rozdelenia obrázku na subregióny. Tieto regióny sú skúmané na nižších rozlíšeniach alebo slabších klasifikátoroch, aby mohli byť rýchlo odpísané, ak tam určite nie je tvár. V každej ďalšej fáze sa subregióny, ktoré postúpili ako potenciálna tvár, skúmajú vo vyšších rozlíšeniach / silnejších klasifikátoroch a odmietajú sa ďalšie subregióny. Nakoniec zostanú len regióny s tvárou.

### 4.3 Extrahovanie príznakov

Vo väčšine odborných článkov a literatúre zameranej na rozpoznávanie ľudských tvárí sa autori venujú hlavne extrakcii príznakov a ich následnému porovnávaniu. Extrakcia príznakov je považovaná za najdôležitejšiu a najdiskutovanejšiu úlohu rozpoznávania ľudí podľa tváre. Z dôvodu zníženia výpočtovej a časovej náročnosti, a zároveň zjednodušenia procesu rozpoznávania sa snažíme extrahovať iba príznaky s najvyššou informačnou hodnotou, ktoré najlepšie charakterizujú hľadaný objekt alebo osobu. Existujúce metódy extrakcie príznakov sa dajú rozdeliť do troch základných kategórií:

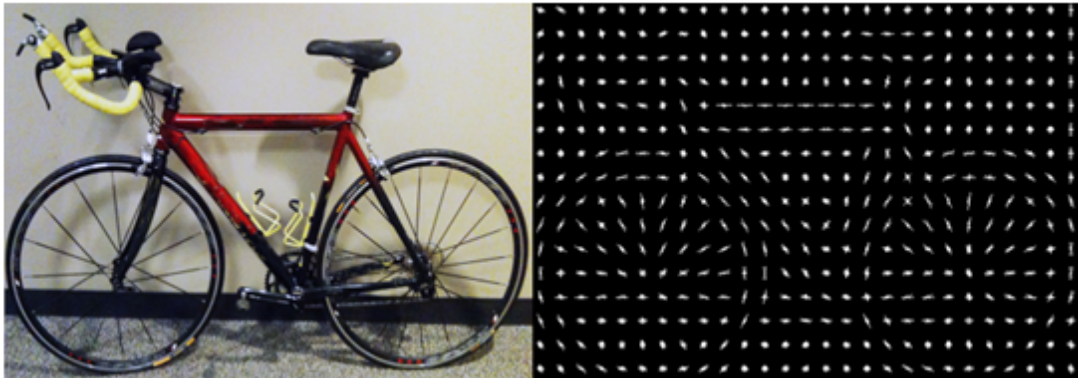
- Holistické metódy – Holistický prístup znamená celostný pohľad na systém, tieto metódy tým pádom extrahujú príznaky z celého vstupného obrázku. Výstupom takýchto metód je jeden viacrozmerný vektor príznakov pre každý vstupný obrázok. Vďaka celostnému pohľadu na obrázky sú v danom vektore obsiahnuté všetky potrebné informácie o danom obrázku. Keďže pracujú s veľkým množstvom dát je rozpoznávanie pamäťovo a výpočtovo náročné (Např. PCA, LDA).
- Lokálne metódy – Lokálne metódy najprv rozdelia obrázok na men-

šie oblasti a extrahujú z nich lokálne príznaky (napr. tvár, telo, nohy). Každý obrázok je reprezentovaný viacerými vektormi príznakov, ktoré charakterizujú dané lokálne oblasti obrázku. Výhodou takýchto metód extrahovania príznakov je, že aj pri čiastočnej oklúzii istých lokálnych oblastí sa dajú objekty porovnávať iba pomocou zvyšných oblastí, ktoré sú viditeľné (nos je zakrytý, porovnáme pomocou očí a úst). Nevýhodou je nutnosť rozdelenia obrazu na lokálne oblasti, prípadne aj detekcia jednotlivých častí hľadaných objektov. Medzi lokálne metódy patrí napr. SIFT, SURF.

- Hybridné metódy – Hybridné metódy kombinujú holistické a lokálne príznaky pri rozpoznávaní objektov. Snaha je využiť výhody oboch týchto prístupov a získať viac typov informácií o hľadaných objektoch.

#### 4.3.1 Typy príznakov do trénovacej množiny

Na rôzne objekty je možné zvoliť rôzne typy príznakov, `trainCascadeObjectDetector` podporuje tieto tri typy: Haar, local binary patterns (LBP), histograms of oriented gradients (HOG). Haar a LBP príznaky sa často používajú na detekciu tváří na obrázkoch. HOG príznaky sa zase často používajú na detekciu objektov ako sú ľudia alebo autá. Sú užitočné na rozpoznanie celkového tvaru objektu. Napríklad na obr.4.8 je vidieť obrys bicykla získaný pomocou HOG príznakov.



Obr. 4.8: HOG príznaky obrázku bicykla

### 4.3.2 Local Binary Pattern (LBP)

Local Binary Pattern (LBP) príznaky dosahujú dobre výsledky v rôznych aplikáciách. Používajú sa napríklad na klasifikáciu textúry a segmentáciu, alebo kontrolu povrchu. Originálny LBP operátor priradzuje hodnoty pixlom obrázkov na základe porovnávania hodnoty stredného pixlu s jeho 3x3 okolím. Výsledkom tohto porovnania je binárny výsledok 1 ak je hodnota väčšia ako stredný pixel alebo 0 ak je menšia. Obr. 4.9. zobrazuje kalkuláciu LBP príznakov [RAHI13].

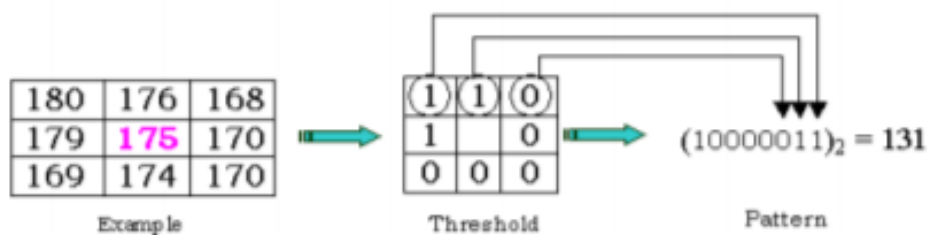
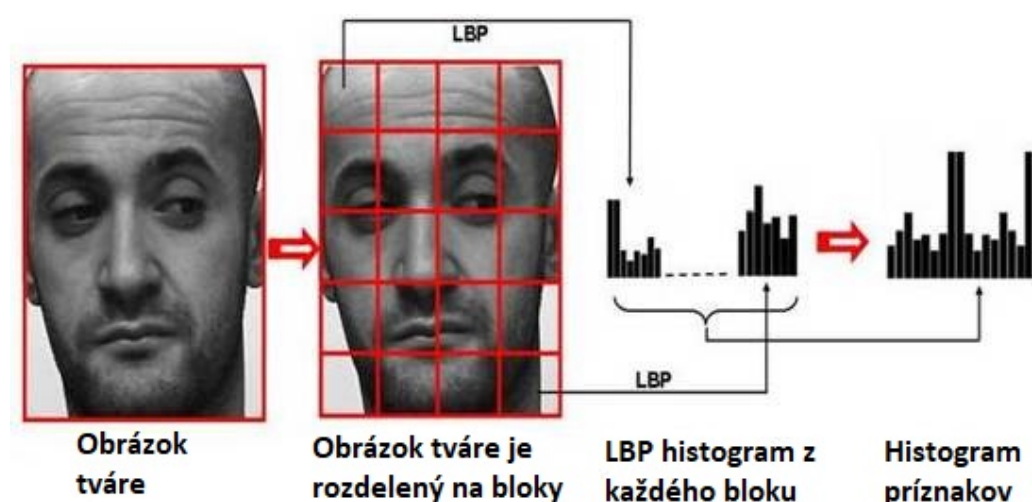


Figure 2. Example of LBP calculation

Obr. 4.9: Ukážka LBP kalkulácie



Lokálne primitíva, ktoré sú kodifikované týmito príznakmi, môžu predstavovať rôzne zaoblené rohy, ploché oblasti a podobne. LBP operátor sa dá prispôbiť na rôzne veľkosti susedných oblastí okrem 3x3. Každý obrázok tváre sa dá chápať ako kompozícia mikro-vzorcov, ktoré sa dajú efektívne detegovať pomocou LBP operátora. Na zakódovanie informácií o týchto vzorcoch v tvári je tento obrázok rozdelený na M malých neprekrývajúcich sa regiónov. LBP histogram extrahovaný z každého z týchto regiónov je potom spojený do jedného priestorovo rozšíreného histogramu. Výsledný histogram reprezentuje lokálnu textúru a celkový tvar tváre. Tento proces je zobrazený na obr. 4.10.



Obr. 4.10: Reprezentácia tváre pomocou LBP

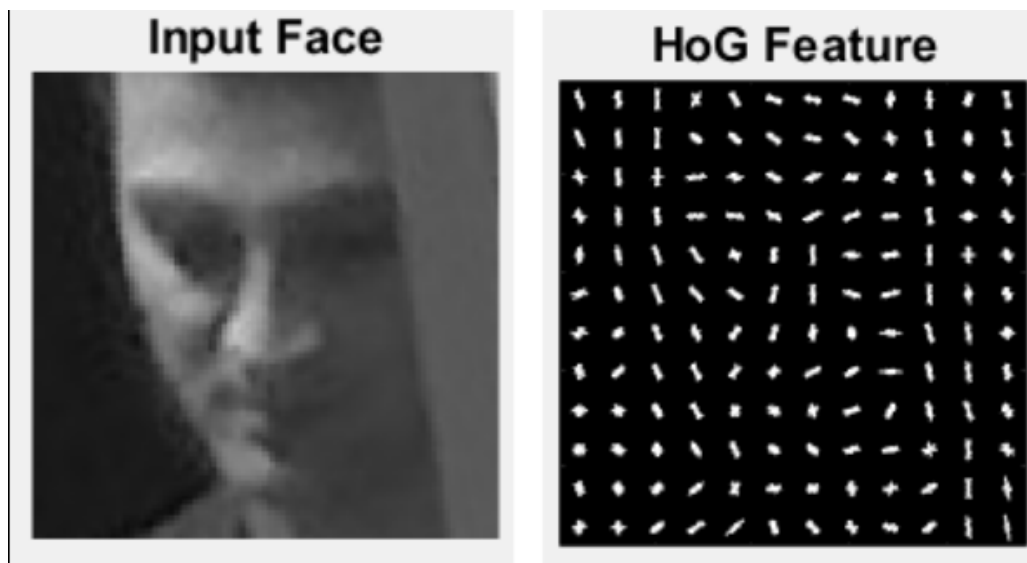
### 4.3.3 Histogram of oriented gradients (HOG)

HOG algoritmus je založený na myšlienke, že tvar a výzor lokálnych objektov v obrázku sa dá opísať distribúciou intenzít gradientov. Gradient je vektor definovaný dvoma premennými: smer a magnitúda. Tieto dve premenné majú iný výpočet, ale obidve sú vypočítané pomocou procesu konvolúcie. Smer

gradientu je kolmý k smeru rohov. Magnitúda gradientu obsahuje informáciu o sile rohu (intenzita pixlov v danom regióne je výrazne rozdielna od okolitých pixlov). Obrázok je rozdelený na malé spojené regióny, tkz. bunky (cells). Pre pixle v každom z týchto buniek je vytvorený histogram smeru gradientov. Deskriptor je vytvorený zlúčením všetkých týchto histogramov.

Keďže HOG deskriptor sa počíta na lokálnych bunkách, je invariantný voči geometrickým a fotometrickým transformáciám, okrem zmenám v orientácii objektu.

HOG deskriptor sa dá použiť pre rozpoznávanie objektov tým, že sa poskytne algoritmom strojového učenia ako príznakový vektor. HOG príznaky sa používajú najmä v spojení s tkz. Support vector machines (SVM) klasifikátorom. HOG príznaky sú zobrazené na obr. 4.11.



Obr. 4.11: Reprezentácia tváre pomocou HOG príznakov

### 4.3.4 Eigenfaces

Eigenfaces [Wag11] je názov pre prístup k rozpoznávaniu tvárí, ktorý je založený na zachytení rozdielov medzi množinou obrázkov tvárí a použití tejto informácie na zakódovanie a porovnávanie jednotlivých tvárí holistickým spôsobom (na rozdiel od spôsobov založených na porovnávaní príznakov alebo regiónov obrázkov). Eigenfaces sú hlavné komponenty (principal components) distribúcie tvárí, respektíve eigenvektory kovariančnej matice množiny obrázkov tvárí. Obrázok s  $N$  pixlami je reprezentovaný ako bod (alebo vektor) v  $N$ -rozmernom priestore. Eigenfaces prístup je mnohými považovaný za prvú funkčnú technológiu rozpoznávania tvárí. Často sa používa ako základná porovnávací metóda na demonštrovanie minimálnej výkonnosti očakávanej od podobných systémov.

Princíp Eigenfaces algoritmu je založený na dvoch úlohách:

1. Extrahuj najrelevantnejšie informácie o tvári, ktoré môžu ale nemusia byť podobné ľudskej intuícii o hlavných príznakoch tváre ako sú oči, nos a pery. Jedným spôsobom ako to dosiahnuť, je zachytenie štatistickej variácie medzi obrázkami tvárí.
2. Reprezentuj obrázok tváre efektívne. Na redukovanie výpočtovej náročnosti a priestorovej komplexnosti môže byť každý obrázok tváre reprezentovaný použitím malej množiny parametrov (príznačky, ktoré sú na každej tvári rovnaké, alebo ich výpovedná hodnota je veľmi malá, sú zahodené).

Eigenfaces sa dajú chápať ako množina príznakov, ktoré charakterizujú globálnu variáciu medzi obrázkami tvárí. Potom je každá tvár aproximovaná použitím podmnožín z eigenfaces, ktoré sú spojené s najväčšími hodnotami

(eigenvalues). Tieto príznaky zahŕňajú najviac variácie v tréningovej množine obrázkov.

Obr. 4.12: Ľavý obrázok v hornom riadku predstavuje priemernú tvár z trénovacej množiny, ďalšie dve obrázky v hornom riadku sú eigenfaces s najvyšším hodnotením. Druhý riadok predstavuje tri najhoršie hodnotené eigenfaces (majú najmenšiu hodnotu eigenvalue).



Obr. 4.12: Ukážka eigenfaces

Opis algoritmu Eigenfaces:

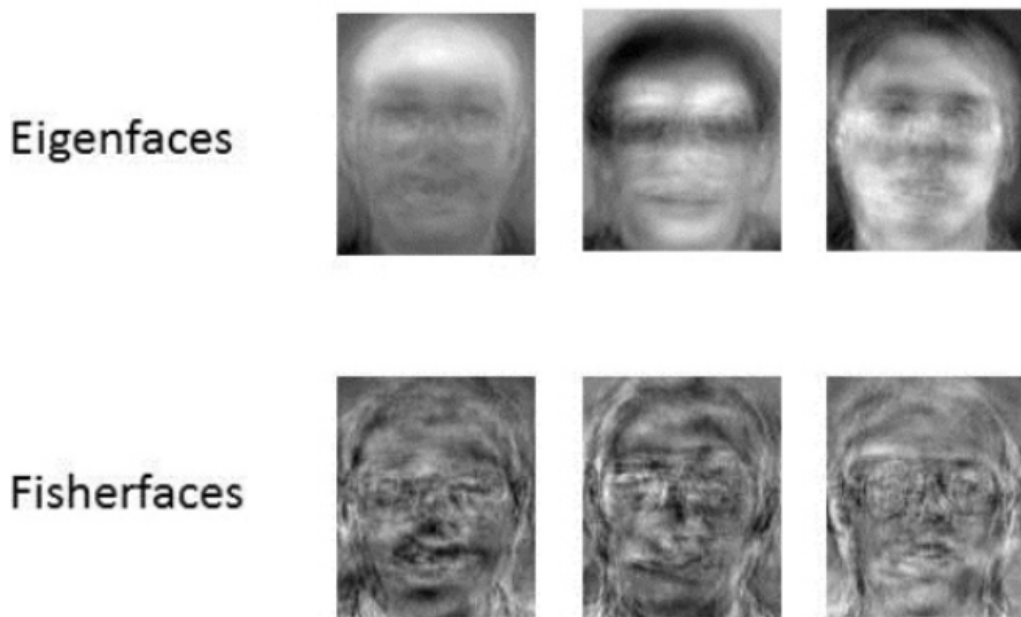
- Príprava matice obrázkov  $X$ , kde každý stĺpec dát predstavuje jeden obrázok. Každý obrázok je priradený tride v korešpondujúcom triednom vektore  $C$ .
- Odčítanie priemerného obrázka z dát.

- Vypočítanie eigenvektorov a eigenhodnôt z kovariančnej matice.
- Nájdenie optimálnej transformačnej maticu vybraním hlavných komponentov (eigenvektory s najvyššou eigenhodnotou).
- Premietnutie centrovaných dát do podpriestoru.

### 4.3.5 Fisherfaces

Kľúčový problém v počítačovom videní, rozpoznávaní obrazcov a strojovom učení spočíva vo vhodnom reprezentovaní dát pre vybraný problém. Jeden spôsob ako reprezentovať vstupné dáta je nájsť podpriestor, ktorý reprezentuje najviac variácie v dátach. Tento podpriestor sa dá získať použitím procedúry s názvom Principal Components Analysis (PCA). Aplikovaním na množinu obrázkov tvárí, PCA vracia množinu premenných s názvom eigenfaces. Tento spôsob reprezentovania dát zaisťuje, že variácia dát je zachovaná a zároveň eliminuje nepotrebné existujúce korelácie medzi originálnymi príznakmi vzorových vektorov.

Ak je úlohou klasifikácia namiesto reprezentácie, je vhodnejšie nájsť podpriestor, ktorý mapuje vzorové vektory rovnakej triedy na jednom mieste príznakovej reprezentácie a vzorové vektory ostatných tried čo najďalej od seba. Techniky na dosiahnutie takéhoto usporiadania sa nazývajú diskriminačné analýzy (discriminant analysis). Najznámejšia takáto technika sa nazýva lineárna diskriminačná analýza (LDA). Keď sa použije LDA (obr. 4.13) na reprezentáciu podpriestoru množiny obrázkov tvárí, výsledné bázové vektory (basis vectors) popisujúce daný priestor sa nazývajú Fisherfaces [Wag12] (obr. 4.14).



Obr. 4.13: Porovnanie eigenfaces a fisherfaces

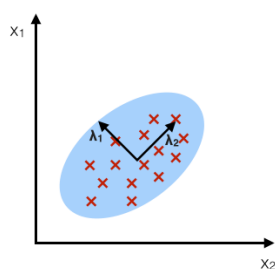
Opis algoritmu Fisherfaces:

- Príprava matice obrázkov  $X$ , kde každý stĺpec dát predstavuje jeden obrázok. Každý obrázok je priradený tride v korešpondujúcom triednom vektore  $C$ .
- Premietnutie dát do  $(N - c)$  rozmerného podpriestoru  $P$  pomocou rotačnej matice  $WP_{ca}$  získanej pomocou PCA, kde:
  - $N$  je počet vzoriek vo vektore  $X$ ,
  - $c$  je počet unikátnych tried ( $length(unique(C))$ ).
- Vypočítanie rozptylu medzi triedami podpriestoru  $P$  ako  $Sb = \sum_{i=1}^c N_i * (mean_i - mean) * (mean_i - mean)^T$ , kde
  - $mean$  je celkový priemer projekcie  $P$ ,

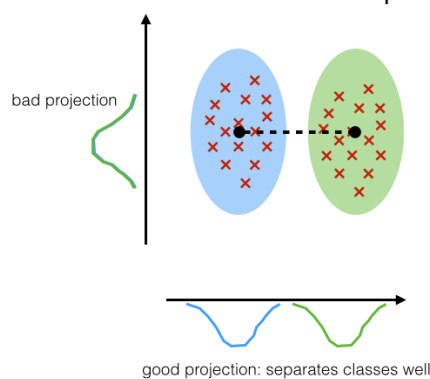
- $mean_i$  je priemer triedy  $i$  v projekcii  $P$ ,
- $N_i$  je počet vzoriek triedy  $i$  v projekcii  $P$ .
- Vypočítanie rozptylu v triedach podpriestoru  $P$  ako  $Sw = \sum_{i=1}^c \sum_{x_k \in X_i} (x_k - mean_i) * (x_k - mean_i)^T$ , kde
  - $X_i$  sú vzorky matice  $X$ ,
  - $x_k$  je vzorka z  $X_i$ ,
  - $mean_i$  je priemer triedy  $i$  v projekcii  $P$ .
- Použitie LDA a maximalizovanie rozptylu medzi triedami a vnútri tried. Riešenie je dané množinou eigenvektorov  $W_{fld}$  z  $S_b$  a  $S_w$  priliehajúcich ich eigenhodnote. Úroveň  $S_b$  je najviac  $(c-1)$ , čiže je iba  $(c-1)$  nenulových eigenhodnôt, zvyšok sa odreže.
- Získanie fisherfaces príznakov ako  $W = W_{Pca} * W_{fld}$ .

**PCA:**

component axes that maximize the variance

**LDA:**

maximizing the component axes for class-separation



Obr. 4.14: Rozdiel medzi PCA (eigenfaces) a LDA (fisherfaces) rozdelením priestoru [Ras14]

## 4.4 Klasifikácia príznakov

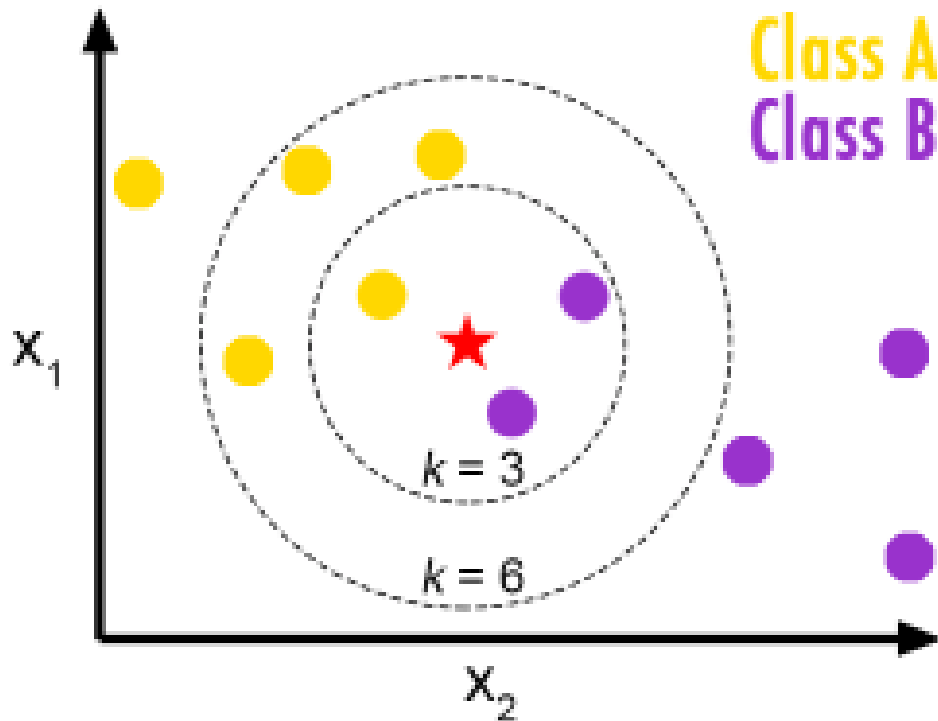
### 4.4.1 K-nearest neighbors (KNN)

Metóda klasifikácie pomocou KNN je neparametrická. To ju robí jednoduchou, ale často efektívnou metódou. Aby sa dátový záznam  $t$  mohol klasifikovať, zistí sa jeho  $k$  (premenná) najbližších susedov a vytvorí sa susedstvo pre záznam  $t$ . Pomocou väčšinového hlasovania spomedzi dátových záznamov obsiahnutých v susedstve záznamu  $t$  sa určí klasifikácia záznamu  $t$ . Aby sme boli schopní efektívne použiť KNN klasifikátor, je treba vhodne zvoliť hodnotu  $k$ , keďže úspešnosť klasifikátora väčšinou veľmi závisí na tejto hodnote. V podstate, KNN metóda je skreslená podľa hodnoty  $k$ . Existuje veľa rôznych spôsobov určovania hodnoty  $k$ . Najjednoduchší spôsob z nich je viacnásobné otestovanie algoritmu pre rôzne hodnoty  $k$  a určenie najlepšieho výsledku, tým pádom najvhodnejšiu hodnotu  $k$  [GWB<sup>+</sup>03].

Ide o metódu učenia s učiteľom, kde sa klasifikujú prvky reprezentované viacdimenziálnymi vektormi do dvoch alebo viacerých tried. Vo fáze učenia sa predspracuje trénovacia množina tak, aby všetky príznaky mali strednú hodnotu 0 a rozptyl 1. To umiestni každý prvok trénovacej množiny do niektorého miesta v  $N$ -rozmernom priestore. Vo fáze klasifikácie sa potom umiestni dátový prvok do triedy na základe najpočetnejšej triedy v jeho susedstve v danom priestore (obr. 4.15).

Ak sa hodnota  $k$  rovná 1, jedná sa o špeciálny zjednodušený prípad, metódu najbližšieho suseda.



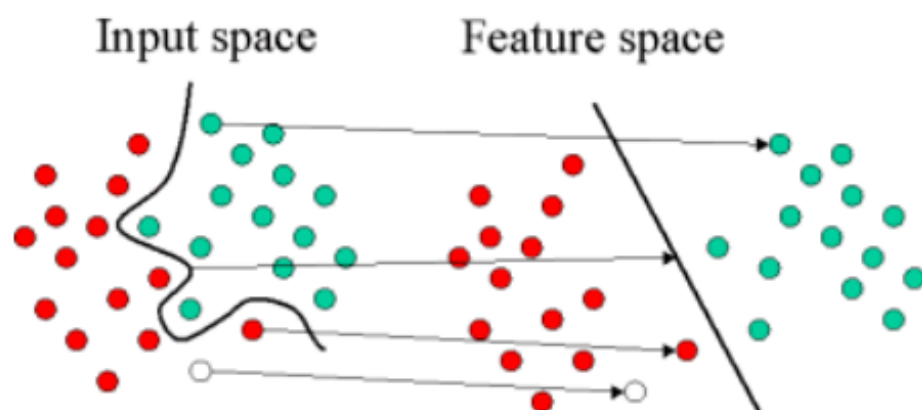
Obr. 4.15: Klasifikácia podľa hodnoty  $k$  [Vai16]

#### 4.4.2 Support Vector Machines (SVM)

SVM klasifikátor je založený na koncepte rozhodujúcich rovín, ktoré predstavujú hranice jednotlivých tried. Rozhodujúca rovina delí množinu objektov, ktoré prilieliajú k rôznym triedam. Ukážka rozdelenia je znázornená na (obr. 4.16). V tejto ilustrácii je znázornená základná myšlienka SVM klasifikátorov. Vidíme v nej originálne objekty (ľavá strana schémy) zmapované, resp. usporiadané pomocou matematických funkcií s názvom kernel. Proces usporiadania objektov sa nazýva mapovanie (transformácia). V tomto novom usporiadaní je množina zmapovaných objektov (pravá strana schémy) lineárne separovateľná. Vďaka tomu nemusíme konštruovať zložité línie. Stačí nájsť

optimálnu priamku, ktorá rozdelí zelené a červené objekty. Táto priamka sa snaží rozdeliť priestor tak, že prvky jednotlivých tried sú od seba čo najďalej vzdialené [Sta18].

SVM sa pri analyzovaní dát používa na klasifikáciu a taktiež na regresívnu analýzu.



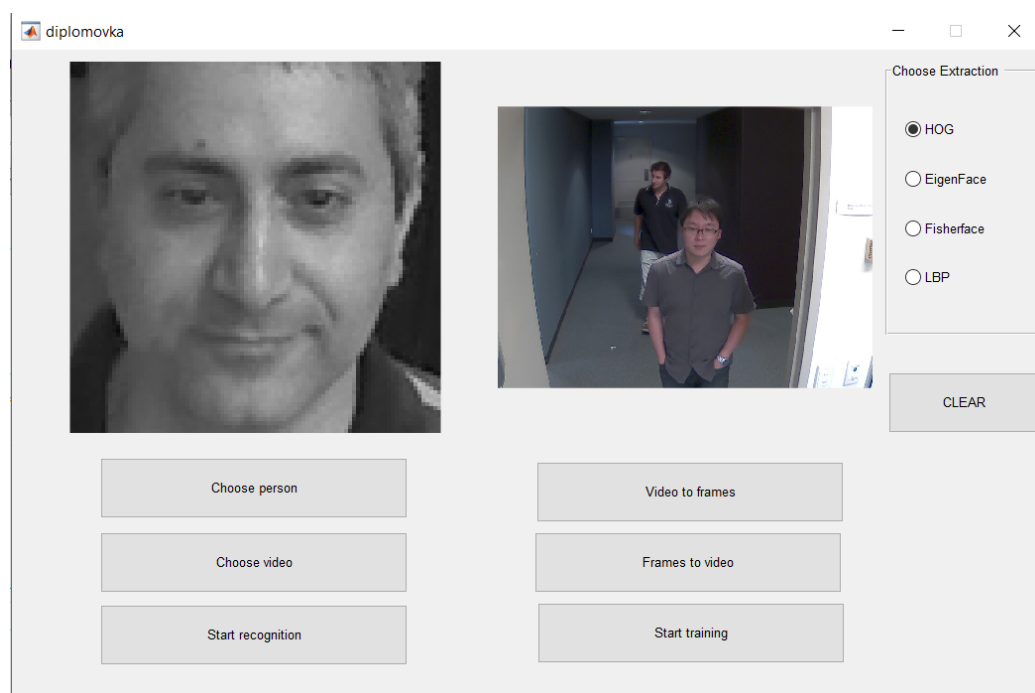
Obr. 4.16: Rozdelenie do tried podľa SVM [Sta18]

# Kapitola 5

## Návrh riešenia

### 5.1 Návrh používateľského rozhrania (GUI)

Rozpis funkcionality používateľského rozhrania na obr. 5.1.



Obr. 5.1: Používateľské rozhranie

- Choose person - Výber hľadanej osoby (dialóg)
- Choose video - Výber videa, v ktorom má algoritmus hľadať vybranú osobu (dialóg)
- Start recognition - Spustí vyhľadávanie vybranej osoby vo vybranom videu
- Video to frames - Premena a uloženie vybraného videa na snímky do vytvoreného priečinka (dialóg)
- Frames to video - Premena a uloženie vybraného priečinka snímok na video do vytvoreného priečinka (dialóg)
- Start training - Spustí natréňovanie klasifikátora podľa vybraného spôsobu extrahovania príznakov
- Choose Extraction - Zaklikávací box možností extrahovania príznakov
- Clear - vyčistenie a zatvorenie všetkých axis a figures (vygenerovaných obrázkov)
- Ľavý obrázok - profil hľadanej osoby
- Pravý obrázok - snímky z vybraného videa

## 5.2 Postup algoritmu rozpoznávania

1. Výber databázy osôb na natréňovanie klasifikátora.
2. Výber spôsobu extrahovania príznakov z detegovaných tvárí osôb z dostupných možností (LBP, HOG, Fisherface, Eigenface, SURF,...).
3. Natréňovanie klasifikátora

4. Výber hľadanej osoby
5. Výber videa, v ktorom má algoritmus hľadať vybranú osobu
6. Prechádzanie snímok videa a detegovanie tvárí pomocou zvoleného detektora.
7. Extrahovanie príznakov z detegovaných tvárí pomocou spôsobu zvoleného v kroku 2.
8. Predikcia detegovaného človeka pomocou natrénovaného klasifikátora.
9. Ak nie je detegovaná hľadaná osoba, prechádzajú sa ďalšie snímky. Ak je detegovaná hľadaná osoba, kontroluje sa správnosť detekcie (nachádza sa hľadaná osoba aj v nasledujúcich snímkach?)
10. Ak sa hľadaná osoba nájde a prejde kontrolou správnosti detekcie, algoritmus sa zastaví a vyhodí pozitívny výsledok hľadania. Inak vyhodí na konci negatívny výsledok hľadania.

# Kapitola 6

## Implementácia

### 6.1 Implementácia detegovania tvárí

Aplikácia vyhľadávania konkrétnej osoby vo videu podľa vizuálneho vzoru bola naprogramovaná v prostredí Matlab R2017a. V konečnej verzii aplikácie sa detegovanie tvárí vykonáva pomocou Viola-Jones algoritmu, ktorý sa z dvoch testovaných modelov detekcie tvárí (DPM a Viola-Jones) ukázal byť vhodnejší. Výsledky testovania a zdôvodnenie tohto výberu sa nachádzajú v kapitole 7. Na problémy detegovania aj extrahovania príznakov sa v prostredí matlab využíva knižnica Computer Vision System Toolbox. Táto knižnica obsahuje okrem množstva iných algoritmov aj algoritmus Viola-Jones, ktorý sa dá zavolať ako:

```
faceDetector = vision.CascadeObjectDetector('MinSize',[50,50]);
```

Detektor `vision.CascadeObjectDetector` v základnej konfigurácii bez ďalších parametrov slúži na detegovanie tvárí pomocou Haarových príznakov. V našom prípade sú to tváre minimálnej veľkosti 50x50 pixlov. Pridaním parametrov ako napr. 'UpperBody', 'Mouth', 'Nose' a 'EyePairBig' sa tento

detektor dá použiť aj na detegovanie vrchných častí tela, úst, nosov a očí. Po detegovaní tváre, vložení boxu ohraničujúceho oblasť tváre a orezaní tohto boxu nasleduje normalizácia detegovaných tvárí:

```
handles.img2 = imcrop(Im, bbox(i, :));  
scaleFactor = 96/size(handles.img2, 1);  
handles.img2 = imresize(handles.img2, scaleFactor);  
handles.img2 = rgb2gray(handles.img2);
```

Takto orezaný obrázok je škálovaný na veľkosť 96x96 pixlov a prevedený z farebného obrázka na šedotónový, tak ako sú uložené obrázky v použitej databáze tvárí. Po detegovaní a normalizácii upravené obrázky pokračujú ďalej do fázy extrahovania príznakov a rozpoznávania osôb.

## 6.2 Implementácia extrahovania príznakov a rozpoznávania

Po detekcii tvárí a normalizácii sa z takto upravených obrázkov získaná vektor príznakov charakterizujúci danú tvár. Tento vektor môže obsahovať príznaky jedného zo štyroch implementovaných typov na základe výberu. Sú to príznaky typu HOG, LBP, Eigenface a Fisherface. Vyberie sa hľadaná osoba vo videu pomocou dialógového okna. Táto osoba sa vyberá z trénovacej databázy osôb. Nasledujúci proces je rovnaký pre všetky štyri výbery aj keď sa kód líši podľa výberu. Extrahujú sa príznaky z normalizovaného obrázka detegovanej tváre:

```
queryFeatures = extractHOGFeatures(handles.img2);
```

Získa sa totožnosť osoby na obrázku pomocou predikcie natrénovaného klasifikátora:

```
personLabel = predict(faceClassifier, queryFeatures);
```

Nájde sa umiestnenie predikovanej osoby v databáze:

```
booleanIndex = strcmp(personLabel, personIndex);  
integerIndex = find(booleanIndex);
```

Získajú sa stringy pre osobu predikovanú klasifikátorom a hľadanú osobu:

```
personLabel = string(personLabel); (predikovaná osoba)  
person = string(person); (hľadaná osoba)
```

Porovnajú sa tieto dva získané stringy. Ak sa rovnajú, zvýši sa pomocné počítadlo o jeden:

```
subplot(1, 2, 1); imshow(handles.img2); title('QueryFace'); (zobrazenie de-  
tegovanej tváre)
```

```
if strcmp(personLabel, person)  
found = found + 1
```

```
subplot(1, 2, 2); imshow(read(faceDatabase(integerIndex), 1)); title('Found');  
(zobrazenie nájdenej osoby)
```

Ak počítadlo dosiahne hodnotu 5, čiže osoba bola 5 krát nájdená vo videu, zobrazí sa snímok videa kde sa hľadaná osoba nachádza a zastaví sa



prehľadávanie videa:

```
if found == 5  
axes(handles.axes2);  
imshow(Im); title('Found');
```

Trénovanie klasifikátora prebieha načítaním databázy z disku:

```
faceDatabase = imageSet(databasePath, 'recursive');
```

Nasleduje extrahovanie príznakov a natrénovanie klasifikátora, testovaná databáza obsahuje 24 ľudí (jedna trieda pre jedného človeka), kde každý človek v databáze je reprezentovaný 39 obrázkami jeho tváre.

# Kapitola 7

## Výsledky

Výsledky práce boli testované na notebooku s danými parametrami:

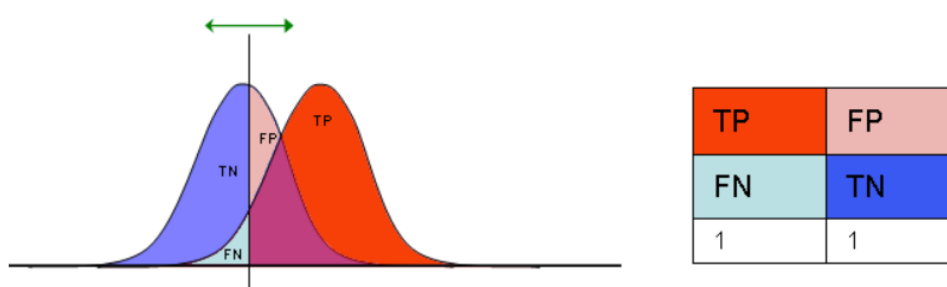
- Procesor - Intel Core i7 7700HQ Kaby Lake
- RAM - 16 GB DDR4
- Grafická karta – NVIDIA GeForce GTX 1060 6 GB MaxQ.
- Disk1 - SSD PCIe M.2 256 GB (nainštalovaný Matlab)
- Disk2 - HDD 1TB 5400 ot./min (uložená databáza)

Zistené testovacie rýchlosti detekcie a rozpoznávania objektov závisia okrem rozlíšenia obrázkov a počtu ľudí na obrázku taktiež od parametrov zariadení, na ktorých boli testované. Použitie silnejšieho počítača, poprípade clusteru spolu spojených počítačov môže výrazne prispieť k zlepšeniu nameraných hodnôt.

## 7.1 Meranie presnosti

V štatistike, meranie diagnostickej schopnosti binárneho klasifikačného systému sa znázorňuje pomocou tzv. ROC krivky (Receiver operating characteristic curve) [Ako07]. Táto krivka ukazuje vzťah medzi špecifikáciou a senzitivitou daného testu alebo detektoru pre všetky prístupné hodnoty prahu (obr. 7.1).

- tp - true positive (v realite je výskyt pozitívny a test našiel správne pozitívny)
- tn - true negative (v realite je výskyt negatívny a test našiel správne negatívny)
- fp - false positive (v realite je výskyt negatívny a test našiel nesprávne pozitívny)
- fn - false negative (v realite je výskyt pozitívny a test našiel nesprávne negatívny)



Obr. 7.1: Sensitivity a Specificity

**Sensitivity (Recall):**

$$\frac{tp}{tp + fn} = \textit{Sensitivity} \quad (7.1)$$

**False negative rate (FNR):**

$$\frac{fn}{tp + fn} = \textit{FNR} \quad (7.2)$$

**Alebo**

$$\textit{FNR} = 1 - \textit{Sensitivity} \quad (7.3)$$

Sensitivity a False negative rate sú komplement jeden druhého, keďže obidve merania záležia na pozitívnych výskytoch v binárnych meraniach (človek je chorý, objekt sa nachádza na fotke, ...).

Tým pádom súčet Sensitivity a False negative rate dáva dokopy 1 alebo 100 percent.

$$\textit{Sensitivity} + \textit{FNR} = 1 \quad (7.4)$$

**Specificity:**

$$\frac{tn}{tn + fp} = \textit{Specificity} \quad (7.5)$$

**False positive rate (FPR):**

$$\frac{fp}{tn + fp} = \textit{FPR} \quad (7.6)$$

Tabuľka 7.1: Tabuľka ROC

		Reality		Total
		Not found	Found	
Test	Not found	TN	FN	
	Found	FP	TP	
Total		Total negative	Total positive	positive + negative

**Alebo**

$$FPR = 1 - Specificity \quad (7.7)$$

Podobne ako Sensitivity a False negative rate, Specificity a False positive rate sú taktiež komplement jeden druhého. Obidve merania záležia na negatívnych výskytoch v binárnych meraniach (človek nie je chorý, objekt sa nenachádza na fotke, ...). Tým pádom súčet Specificity a False positive rate dáva dokopy 1 alebo 100 percent.

$$Specificity + FPR = 1 \quad (7.8)$$

Vzor ROC tabuľky je znázornený v tabuľke 7.1.

### 7.1.1 Testovacia databáza

Na otestovanie efektívnosti vytvorenej aplikácie na vyhľadávanie konkrétnej osoby na videu je potrebné mať k dispozícii dostatočne veľkú testovaciu databázu. Databáza by mala obsahovať obrázky tváří osôb nafotených spredu, ako aj z rôznych uhlov. Taktiež je treba mať k dispozícii dostatočne veľkú vzorku videí, v ktorých sa nachádzajú nafotené osoby z databázy. Na testovanie výsledkov bola použitá jedna takáto voľne dostupná video databáza, ktorá sa nazýva ChokePoint Dataset [WCM<sup>+</sup>11]. Obsahuje videozáznamy z

chodieb pracoviska v 3 rôznych uhľoch v dvoch lokalitách. Táto databáza simuluje reálne kamerové systémy na pracoviskách.

Obrázky tvári osôb majú rozmer 96x96 pixlov a sú v čiernobielom PGM formáte. Snímky z kamier majú rozmer 800x600 pixlov a sú v JPG formáte.

## 7.2 Výsledky detekcie tváří

### 7.2.1 Detekcia pomocou Viola-Jones

Testované na testovacej množine 100 obrázkov(framov). Celkové výsledky testovania detekcie pomocou Viola-Jones sú znázornené v tabuľke 7.2. Priemerná rýchlosť detekcie tváří vo framoch na testovacej množine 100 framov je 0.3256 sekúnd.

Viola-Jones má veľmi rýchly čas výpočtu - iba 0.3256 sekundy na jednu snímku. Tým sa zaraďuje medzi algoritmy, ktoré by sa dali použiť aj pri real-time systémoch detekcie. Má však pomerne vysokú hodnotu False positive rate, čiže nachádza veľa tváří aj tam, kde sa žiadne nenachádzajú. False negative rate má však na vybranej testovacej množine veľmi dobrý, iba 0.04. Dokázal tým pádom úspešne detegovať takmer všetky tváre. Je však treba poznamenať, že vybraných 100 testovacích obrázkov obsahovalo tváre nasnímané z kamerového systému väčšinou pod uhlom menším ako 40 stupňov. Dá sa predpokladať, že pri väčších uhľoch by boli výsledky horšie. Ukážky detekcie na testovacej množine pomocou Viola-Jones detektora sú znázornené na obr. 7.2 až obr. 7.6.

Tabuľka 7.2: Tabuľka ROC Viola-Jones

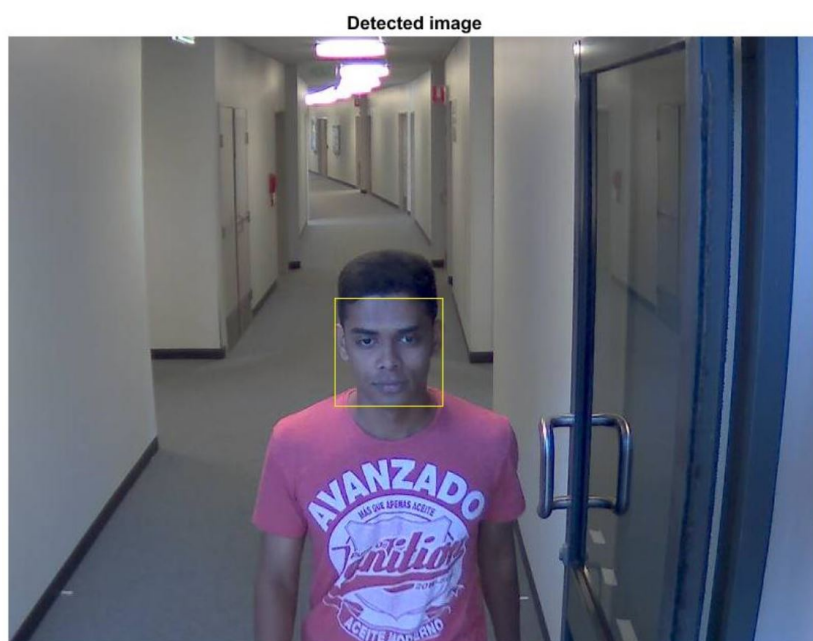
		Reality		
		Not found	Found	
Test	Not found	TN = 18	FN = 4	
	Found	FP = 11	TP = 94	
Total		Total negative = 29	Total positive = 98	
	Sensitivity = 0.96	Specificity = 0.62	FPR = 0.38	FNR = 0.04



Obr. 7.2: Viola-Jones detekcia 1

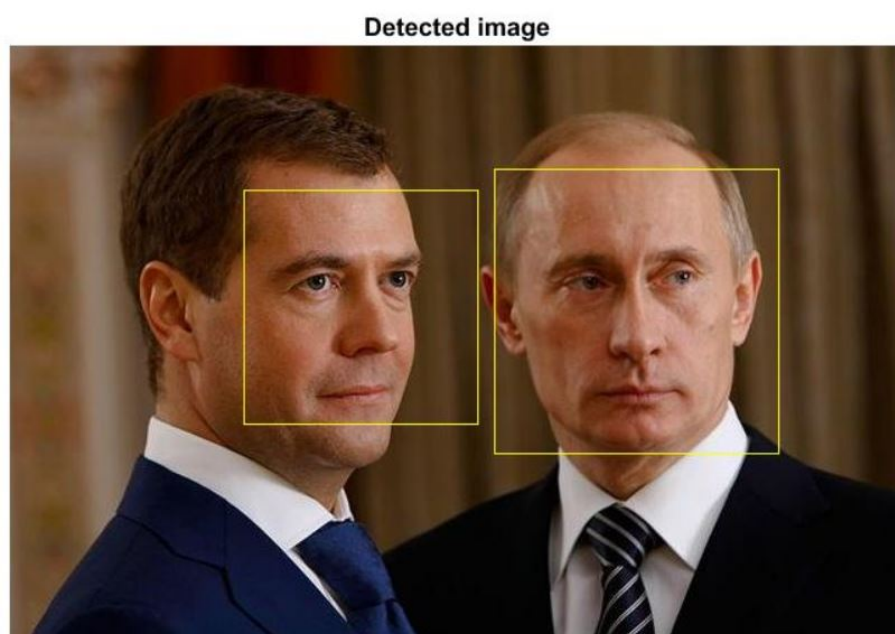


Obr. 7.3: Viola-Jones detekcia 2



Obr. 7.4: Viola-Jones detekcia 3





Obr. 7.5: Viola-Jones detekcia 4



Obr. 7.6: Viola-Jones detekcia 5

Tabuľka 7.3: Tabuľka ROC DPM

		Reality		
		Not found	Found	
Test	Not found	TN = 20	FN = 9	
	Found	FP = 0	TP = 89	
Total		Total negative = 20	Total positive = 98	
	Sensitivity = 0.908	Specificity = 1.0	FPR = 0	FNR = 0.092

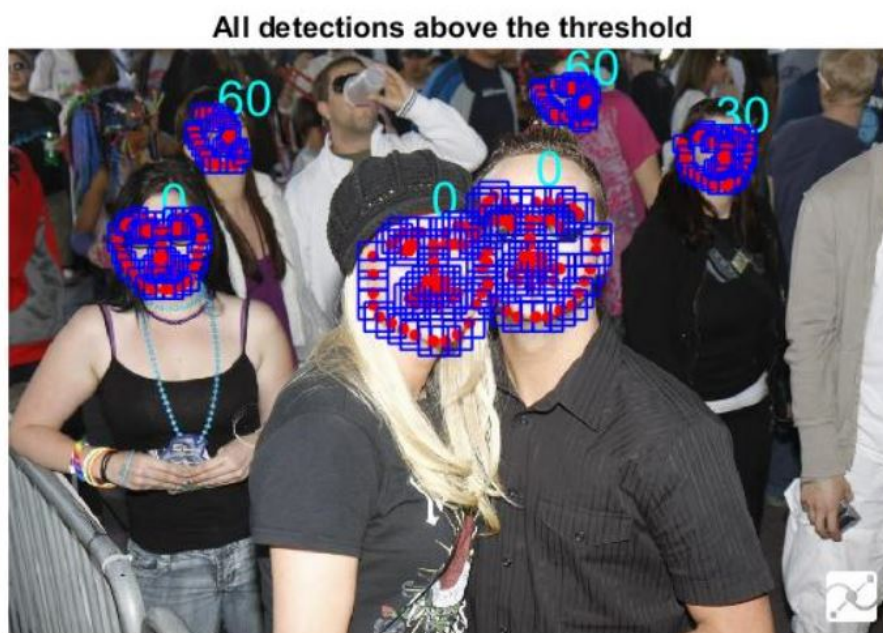
### 7.2.2 Výsledky Deformable parts model (DPM) detekcie

Testované na testovacej množine 100 obrázkov(framov). Celkové výsledky testovania detekcie pomocou DPM sú znázornené v tabuľke 7.3. Priemerná rýchlosť detekcie tvárí vo framoch na testovacej množine 100 framov je 20.438 sekúnd.

DPM má veľmi pomalý čas výpočtu - 20.438 sekundy na jednu snímku. Vďaka vstavanému a nastaviteľnému prahu (thresholdu) detekcie má nulový False positive rate, čiže nemá žiadne false positive detekcie. False negative rate má na vybranej testovacej množine veľmi dobrý, iba 0.092. Dokázal tým pádom úspešne detegovať takmer všetky tváre. Napriek lepším výsledkom než má Viola-Jones algoritmus, som sa rozhodol pri testovaní ďalších fáz pre Viola-Jones algoritmus. Dôvodom je, že Viola-Jones má dostatočnú presnosť detekcie pre kamerové systémy, pri ktorých sa dá predpokladať, že osoba je nasnímaná pod 0 až 40 stupňovým uhlom a zároveň je nepomerne rýchlejší. Ukážky detekcie na testovacej množine pomocou DPM detektora sú znázornené na obr. 7.7 až obr. 7.11.



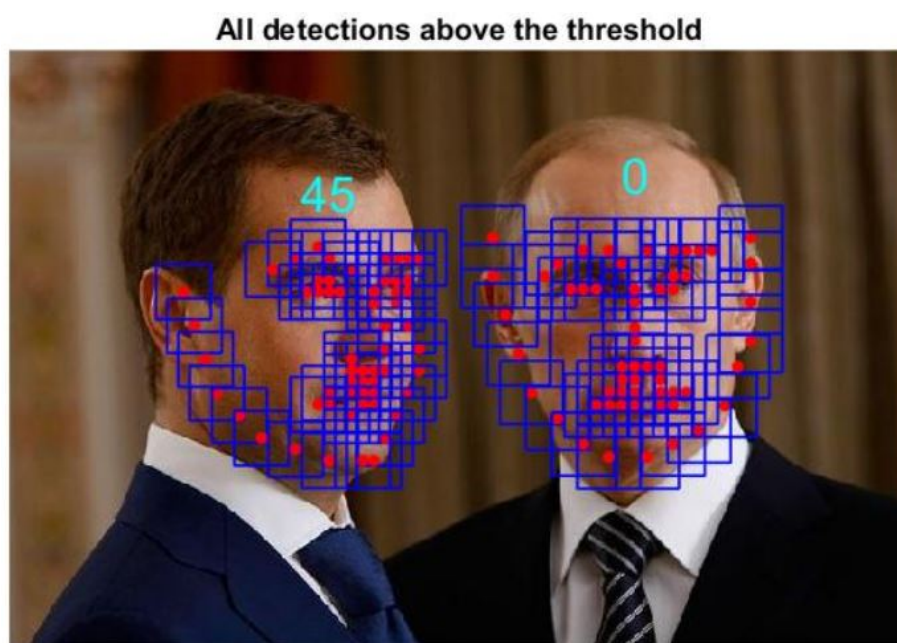
Obr. 7.7: DPM detekcia 1



Obr. 7.8: DPM detekcia 2



Obr. 7.9: DPM detekcia 3



Obr. 7.10: DPM detekcia 4

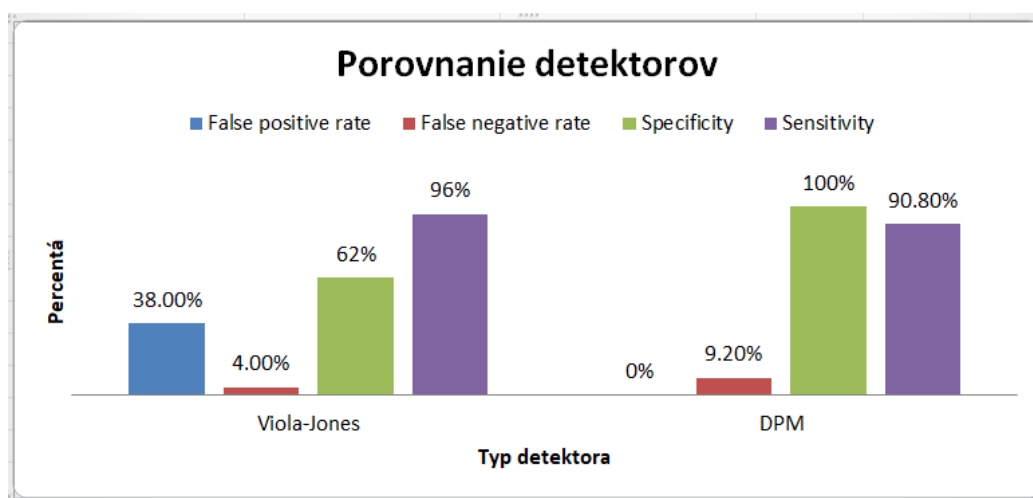


Obr. 7.11: DPM detekcia 5

### 7.2.3 Porovnanie detektorov

Porovnanie výsledkov detekcie osôb pomocou dvoch testovaných detektorov sú znázornené na obr. 7.12





Obr. 7.12: Porovnanie výsledkov detektorov

## 7.3 Výsledky rozpoznávania osôb

Na natrénovanie klasifikátorov boli použité vektory príznakov získané z 39 trénovacích obrázkov pre jednu osobu a celkovo bolo 24 osôb v trénovacej množine.

### 7.3.1 Rozpoznávanie pomocou HOG príznakov

Testované na testovacej množine 183 obrázkov(framov). Rozpoznávanie osôb pomocou HOG príznakov bolo testované natrénovaním SVM a KNN klasifikátorov.

#### HOG príznaky + SVM (Support vector machine) klasifikátor

Celkové výsledky rozpoznávania pomocou HOG+SVM príznakov sú znázornené v tabuľke 7.4.

Priemerný čas natrénovania klasifikátora: 13.4850 sekúnd.

Tabuľka 7.4: Tabuľka ROC HOG + SVM

		Reality		
		Not found	Found	
Test	Not found	TN = 27	FN = 6	
	Found	FP = 2	TP = 138	
Total		Total negative = 29	Total positive = 144	
	Sensitivity = 0.958	Specificity = 0.931	FPR = 0.069	FNR = 0.042

Tabuľka 7.5: Tabuľka ROC HOG + KNN

		Reality		
		Not found	Found	
Test	Not found	TN = 27	FN = 5	
	Found	FP = 3	TP = 138	
Total		Total negative = 30	Total positive = 143	
	Sensitivity = 0.965	Specificity = 0.9	FPR = 0.1	FNR = 0.035

Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0012 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.1277 sekúnd.

### **HOG príznaky + KNN (k-nearest neighbors) klasifikátor**

Celkové výsledky rozpoznávania pomocou HOG+KNN príznakov sú znázorňované v tabuľke 7.5.

Priemerný čas natrénovania klasifikátora: 2.7692 sekúnd.

Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0011 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.0402 sekúnd.

Tabuľka 7.6: Tabuľka ROC LBP + SVM

		Reality		
		Not found	Found	
Test	Not found	TN = 33	FN = 123	
	Found	FP = 0	TP = 17	
Total		Total negative = 33	Total positive = 140	
	Sensitivity = 0.121	Specificity = 1	FPR = 0	FNR = 0.879

### 7.3.2 Rozpoznávanie pomocou LBP príznakov

Testované na testovacej množine 183 obrázkov(framov). Rozpoznávanie osôb pomocou LBP príznakov bolo testované natrénovaním SVM a KNN klasifikátorov.

#### LBP príznaky + SVM (Support vector machine) klasifikátor

Celkové výsledky rozpoznávania pomocou LBP+KNN príznakov sú znázornené v tabuľke 7.6.

Priemerný čas natrénovania klasifikátora: 4.8885 sekúnd.

Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0014 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.1107 sekúnd.

#### LBP príznaky + KNN (k-nearest neighbors) klasifikátor

Celkové výsledky rozpoznávania pomocou LBP+KNN príznakov sú znázornené v tabuľke 7.7.

Priemerný čas natrénovania klasifikátora: 2.9440 sekúnd.

Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0014 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.0048 sekúnd.



Tabuľka 7.7: Tabuľka ROC LBP + KNN

		Reality		
		Not found	Found	
Test	Not found	TN = 33	FN = 109	
	Found	FP = 0	TP = 31	
Total		Total negative = 33	Total positive = 140	
	Sensitivity = 0.221	Specificity = 1	FPR = 0	FNR = 0.779

Tabuľka 7.8: Tabuľka ROC Eigenface + KNN

		Reality		
		Not found	Found	
Test	Not found	TN = 33	FN = 36	
	Found	FP = 0	TP = 104	
Total		Total negative = 33	Total positive = 140	
	Sensitivity = 0.743	Specificity = 1	FPR = 0	FNR = 0.257

### 7.3.3 Rozpoznávanie pomocou Eigenface príznakov

#### Eigenface príznaky + KNN (k-nearest neighbors) klasifikátor

Testované na testovacej množine 183 obrázkov(framov). Rozpoznávanie osôb pomocou Eigenface príznakov bolo testované natrénovaním KNN klasifikátora.

Celkové výsledky rozpoznávania pomocou Eigenface príznakov sú znázornené v tabuľke 7.8.

Priemerný čas natrénovania klasifikátora: 1.8771 sekúnd.

Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0020 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.012702 sekúnd.

Tabuľka 7.9: Tabuľka ROC Fisherface + KNN

		Reality		
		Not found	Found	
Test	Not found	TN = 33	FN = 51	
	Found	FP = 0	TP = 89	
Total		Total negative = 33	Total positive = 140	
	Sensitivity = 0.636	Specificity = 1	FPR = 0	FNR = 0.364

### 7.3.4 Rozpoznávanie pomocou Fisherface príznakov

#### Fisherface príznaky + KNN (k-nearest neighbors) klasifikátor

Testované na testovacej množine 183 obrázkov(framov). Rozpoznávanie osôb pomocou Fisherface príznakov bolo testované natrénovaním KNN klasifikátora.

Celkové výsledky rozpoznávania pomocou Fisherface príznakov sú znázornené v tabuľke 7.9.

Priemerný čas natrénovania klasifikátora: 4.4529 sekúnd.

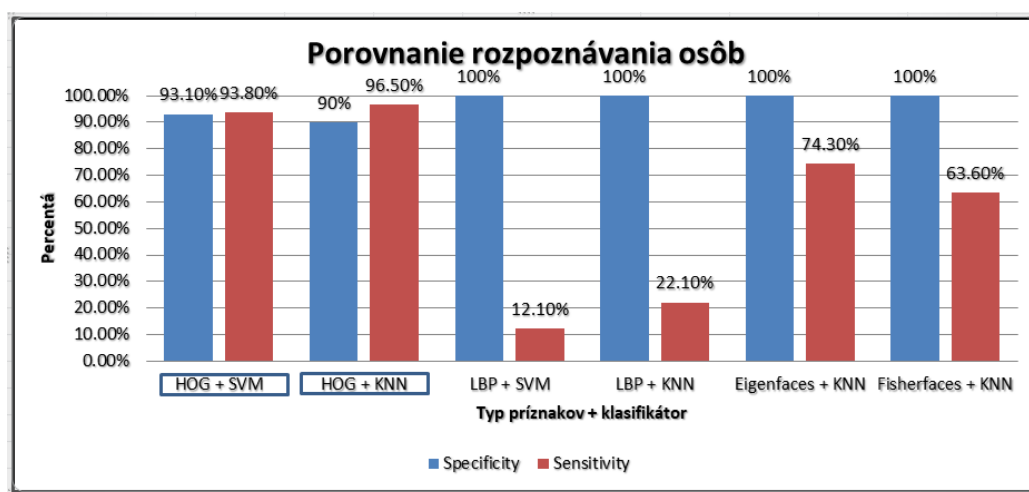
Priemerný čas extrahovania príznakov na jednu detegovanú tvár: 0.0048 sekúnd.

Priemerný čas predikcie osoby na jednu detegovanú tvár: 0.00072414 sekúnd.

### 7.3.5 Porovnanie výsledkov

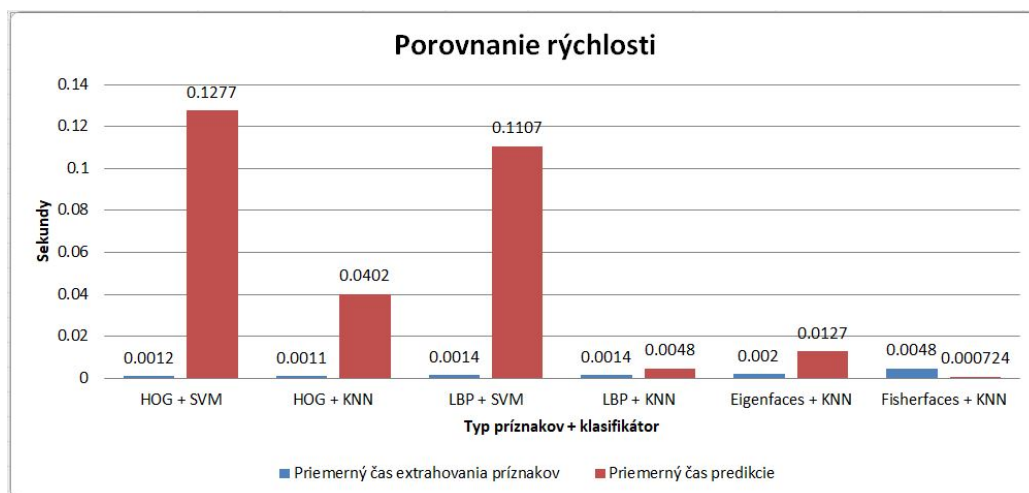
Porovnanie výsledkov rozpoznávania osôb pre rôzne typy príznakov a klasifikátorov sú znázornené na obr. 7.13. Spomedzi testovaných spôsobov extrahovania príznakov sa iba LBP príznaky ukázali byť nevhodné na riešenie daného problému vyhľadávania osôb podľa tváří. Najlepšie výsledky na testovacej množine boli dosiahnuté použitím HOG príznakov v kombinácii s SVM

a j KNN klasifikátorom. LBP príznaky mali najhorší výsledok v kombinácii s obidvoma klasifikátormi. Testovaný bol lineárny SVM klasifikátor. KNN klasifikátor bol testovaný pre hodnoty  $k = 1, 3, 5, 10$ . Najlepšie výsledky boli dosiahnuté pre  $k=5$  a  $k=1$ , pričom rozdiel medzi týmito dvoma hodnotami bol minimálny. Vo finálnej verzii aplikácie a taktiež pri testovaní výsledkov bol použitý KNN klasifikátor s hodnotou  $k=1$ , kvôli jeho rýchlosti výpočtu a dobrým výsledkom predikcie.



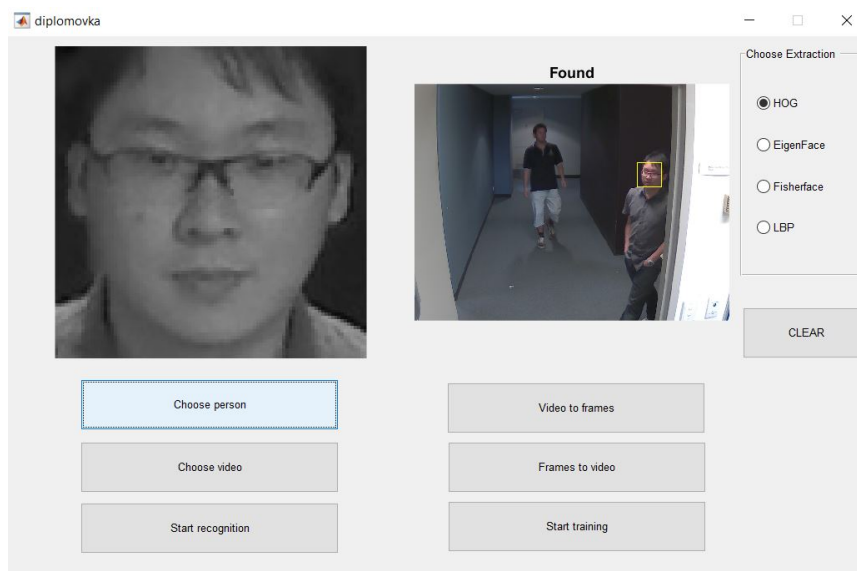
Obr. 7.13: Porovnanie výsledkov príznakov a klasifikátorov

Porovnanie priemerných rýchlostí výpočtu testovaných metód extrahovania príznakov a klasifikátorov sú znázornené na obr. 7.14. Z výsledkov je vidno, že v priemere je extrahovanie príznakov z jednej detegovanej tváre veľkosti 96x96 pixlov rýchlejšie ako predikovanie osoby pomocou vyššie spomenutých klasifikátorov. Celkovo sa ako najpomalšia ukázala kombinácia HOG príznakov s SVM klasifikátorom. Najrýchlejší priemerný čas výpočtu mala naopak kombinácia Fisherface príznakov s KNN klasifikátorom. Všetky metódy však dosahovali celkovú priemernú rýchlosť extrahovania príznakov a predikcie osoby klasifikátorom menšiu ako 0.2 sekundy.



Obr. 7.14: Porovnanie rýchlosti extrahovania a predikcie

Ukážka výsledku vyhľadávania konkrétnej osoby vo videu podľa natrénovaného klasifikátora je vidieť na obr. 7.15. V ľavej časti používateľského rozhrania je zobrazený obrázok hľadanej osoby. V pravej časti je zobrazený snímok videa v ktorom bola hľadaná osoba úspešne rozpoznaná.



Obr. 7.15: Úspešný výsledok vyhľadávania konkrétnej osoby

# Kapitola 8

## Záver

Podarilo sa nám úspešne implementovať rozpoznávanie osôb na ChokePoint databáze pomocou Viola-Jones detekcie tváří a následnej normalizácii na rozmer a typ rovnaký, ako sa nachádza v databáze. Detekcia tváří pomocou DPM modelu sa ukázala byť nevhodná z dôvodu rýchlosti jeho výpočtu. Predikcia osoby sa vykonáva porovnávaním príznakov získanými štyrmi rôznymi spôsobmi extrahovania. Sú to HOG, LBP, Fisherface a Eigenface príznaky. Tieto štyri spôsoby extrahovania boli vybrané na základe naštudovaných materiálov. Eigenface príznaky sú jedným zo štandardných metód rozpoznávania tváří. Použitie fisherface príznakov slúži na porovnanie výsledkov predikcie zmenou rozdelenia priestoru pomocou LDA namiesto PCA. Predikcia pomocou HOG a LBP príznakov je experiment, či sa dajú efektívne rozpoznávať tváre aj pomocou jednoduchších metód, ktoré sa bežne používajú na rozpoznávanie objektov.

Rýchlosťou výpočtu boli všetky spôsoby extrahovania príznakov a predikcie klasifikátorom priemerne v rozmedzí 0.2 sekundy jeden od druhého. Zároveň mali všetky rýchlosť výpočtu menej ako 0.2 sekundy na jeden obrázok detegovanej tváre.

Rozpoznávanie osôb pomocou HOG, Fisherface a Eigenface príznakov a predikcie pomocou natrénovaného klasifikátora sú schopné rozpoznať konkrétneho človeka z testovaných videozáznamov. Základným predpokladom však je, že sa výzor hľadaných osôb výrazne nezmenil od ich výzoru na obrázkoch v testovacej množine. Jedine LBP príznaky sa ukázali byť nevhodné pre riešenie daného problému, bez ohľadu na použitý typ klasifikátora.

# Literatúra

- [Ako07] Anthony K Akobeng. Understanding diagnostic tests 3: receiver operating characteristic curves. *Acta paediatrica*, 96(5):644–647, 2007.
- [Ban14] Jozef Ban. ROZPOZNÁVANIE ĽUDSKÝCH TVÁRÍ. volume 1, pages I–I. FEI, 2014.
- [BP16] Ritesh Boda and M Jasmine Pemeena Priyadarsini. Face detection and tracking using klt and viola jones. *ARPJN Journal of Engineering and Applied Sciences*, 11(23):13472–13476, 2016.
- [GWB<sup>+</sup>03] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *OTM Confederated International Conferences On the Move to Meaningful Internet Systems*, pages 986–996. Springer, 2003.
- [MBPVG14] Markus Mathias, Rodrigo Benenson, Marco Pedersoli, and Luc Van Gool. Face detection without bells and whistles. In *European Conference on Computer Vision*, pages 720–735. Springer, 2014.
- [NK16] Nirvair Neeru and Lakhwinder Kaur. Modified sift descriptors

- for face recognition under different emotions. *Journal of Engineering*, 2016, 2016.
- [RAHI13] Md Abdur Rahim, Md Shafiul Azam, Nazmul Hossain, and Md Rashedul Islam. Face recognition using local binary patterns (lbp). *Global Journal of Computer Science and Technology*, 2013.
- [Ras14] Sebastian Raschka. Linear discriminant analysis, 2014.
- [Sta18] Statsoft. Support vector machines (svm), 2018.
- [TYRW14] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.
- [Vai16] Piush Vaish. k-nearest neighbors(knn), 2016.
- [VJ01] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.
- [Wag11] Philipp Wagner. Eigenfaces, 2011.
- [Wag12] Philipp Wagner. Fisherfaces, 2012.
- [WCM<sup>+</sup>11] Yongkang Wong, Shaokang Chen, Sandra Mau, Conrad Sanderson, and Brian C. Lovell. Patch-based probabilistic image quality assessment for face selection and improved video-based



- face recognition. In *IEEE Biometrics Workshop, Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 81–88. IEEE, June 2011.
- [ZR12] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2879–2886. IEEE, 2012.

# Zoznam obrázkov

4.1	Model rozpoznávania tvárí [Ban14]	5
4.2	Kamerový systém ChokePoint databázy [WCM <sup>+</sup> 11]	8
4.3	Haarov príznak, ktorý sa podobá okoliu očí aplikovaný na ob- rázok	9
4.4	Haarov príznak, ktorý sa podobá nosu aplikovaný na obrázok	9
4.5	Ukážka Viola-Jones detekcie	10
4.6	DPM model tváre v rôznych natočeniach s HOG príznakmi [ZR12]	11
4.7	Porovnanie detektorov tvárí [MBPVG14]	12
4.8	HOG príznaky obrázku bicykla	15
4.9	Ukážka LBP kalkulácie	15
4.10	Reprezentácia tváre pomocou LBP	16
4.11	Reprezentácia tváre pomocou HOG príznakov	17
4.12	Ukážka eigenfaces	19
4.13	Porovnanie eigenfaces a fisherfaces	21
4.14	Rozdiel medzi PCA (eigenfaces) a LDA (fisherfaces) rozdele- ním priestoru [Ras14]	22
4.15	Klasifikácia podľa hodnoty k [Vai16]	24
4.16	Rozdelenie do tried podľa SVM [Sta18]	25

5.1	Používateľské rozhranie . . . . .	26
7.1	Sensitivity a Specificity . . . . .	34
7.2	Viola-Jones detekcia 1 . . . . .	38
7.3	Viola-Jones detekcia 2 . . . . .	39
7.4	Viola-Jones detekcia 3 . . . . .	39
7.5	Viola-Jones detekcia 4 . . . . .	40
7.6	Viola-Jones detekcia 5 . . . . .	40
7.7	DPM detekcia 1 . . . . .	42
7.8	DPM detekcia 2 . . . . .	42
7.9	DPM detekcia 3 . . . . .	43
7.10	DPM detekcia 4 . . . . .	43
7.11	DPM detekcia 5 . . . . .	44
7.12	Porovnanie výsledkov detektorov . . . . .	45
7.13	Porovnanie výsledkov príznakov a klasifikátorov . . . . .	50
7.14	Porovnanie rýchlosti extrahovania a predikcie . . . . .	51
7.15	Úspešný výsledok vyhľadávania konkrétnej osoby . . . . .	51

# Príloha

DVD príloha so zdrojovým kódom a textovými súbormi:

- `Diplomova_praca_kod` - obsahuje zdrojový kód
- `Diplomova_praca_text` - obsahuje textové súbory