Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Rozpoznávanie tvárí

Bakalárska práca

2020
Marek Jaroš

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Rozpoznávanie tvárí

Bakalárska práca

Študijný program:	Aplikovaná informatika
Študijný odbor:	Aplikovaná informatika
Školiace pracovisko:	Katedra informatiky
Školiteľ:	RNDr. Andrej Lúčny, PhD.

Bratislava, 2020
Marek Jaroš

Zadanie práce

Tu bude zadanie bakalárskej práce ...

Poďakovanie: Tu môžete poďakovať školiteľovi, prípadne ďalším osobám, ktoré vám s prácou nejako pomohli, poradili, poskytli dáta a podobne.

Abstrakt

Slovenský abstrakt v rozsahu 100-500 slov, jeden odstavec. Abstrakt stručne sumarizuje výsledky práce. Mal by byť pochopiteľný pre bežného informatika. Nemal by teda využívať skratky, termíny alebo označenie zavedené v práci, okrem tých, ktoré sú všeobecne známe.

Kľúčové slová: jedno, druhé, tretie (prípadne štvrté, piate)

Abstract

Abstract in the English language (translation of the abstract in the Slovak language).

Keywords:

Obsah

Úvod **1**

1 Východiská **3**

1.1 Prehľad problematiky 3

1.1.1 Detekcia tváre 3

1.1.2 Rozpoznávanie tvárí 4

1.2 Metódy pre rozpoznávanie tvárí 6

1.2.1 Eigenfaces 7

1.2.2 Local Binary Patterns Histogram 7

1.3 Použité technológie 10

1.3.1 Python 10

1.3.2 OpenCV 10

1.3.3 Dlib 10

Zoznam obrázkov

1.1 Ukážka Haarového príznaku 4
1.2 Technológia Face ID 5
1.3 Ukážka fungovania DeepFace 6
1.4 Ukážka trénovacej množiny pre Eigenfaces 8
1.5 Ukážka získavania novej hodnoty pri LBPH 9
1.6 Ukážka získavania histogramu pri LBPH 9

Úvod

Cieľom tejto bakalárskej práce je naštudovať si rozpoznávanie tváre za pomoci motódy Local Binary Pattern Histograms a taktiež pomocou hlbokých neurónových sietí (ResNet) trénovaných pomocou metric loss function.

Následne tieto metódy re-implementujeme v programovacom jazyku Python, za pomoci knižníc OpenCV a Dlib. Budeme porovnať ich výkon a analyzovať ich slabiny s ohľadom na rozpoznanie osoby na videu. V neskoršej fáze bakalárskej práce budeme skúmať aktuálne výsledky v problematike rozpoznávania tvárí a porovnáme ich s výkonom nami vyskúšaných metód.

Kapitola 1

Východiská

V tejto kapitole si povieme niečo o histórií deep learningu, predstavíme si problematiku rozpoznávania tvárí a ich využitie. Bližšie si predstavíme dve metódy na rozpoznávanie tvárí - Local Binary Pattern Histograms a metódu za pomoci hlbokých neurónových sietí (ResNet) trénovaných pomocou metric loss function.

Taktiež si problížime technológie, s ktorými budeme v rámci riešenia tejto bakalárskej práce pracovať.

1.1 Prehľad problematiky

V dnešnej dobe, dostatočným výpočtovým výkonom disponujú už aj tak malé a prenosné zariadenia, ako sú naše mobilné telefóny. Aj preto sme v posledných rokoch svedkom javu, kedy sa do týchto zariadení inštaluje systém, ktorý nám ho umožňuje otvoriť len za pomoci našej tváre.

Vďaka tomu sa tak dnes vieme prihlasovať nie len do našich telefónov, ale si napríklad aj z pohodlia domova otvoriť bankový účet. Bez toho, aby sme boli na pobočke banky fyzicky prítomný. Stačí odfotiť svoj občiansky preukaz, následne odfotiť seba a systém okrem odčítania osobných údajov aj automaticky rozpozná, či sa na danom preukaze skutočne nachádza daná osoba.

A to je len jeden z mnohých príkladov, kde sa v bežnom živote stretávame s rozpoznávaním tváre. Avšak, na začiatku našej práce by sme mali rozlíšiť medzi detekciou tváre a jej rozpoznávaním.

1.1.1 Detekcia tváre

Najzákladnejším, no veľmi dôležitým krokom pre rozpoznávanie tváre, je jej detekcia na obrázku či vo videu. O dôležitosti tohto kroku v systémoch pre rozpoznávanie tvárí svedčí aj to, že ich detekcia je problematika, ktorá sa skúma sama o sebe.

Avšak vo všeobecnosti, detekcia tváre spadá do odboru detekcie objektov triedy . Jedným z najpoužívanejších algoritmov pre detekciu tváre je aj ten, ktorý v roku 2001 vynašli Paul Viola a Michael Jones. Jeho meno je **Viola-Jones**. A hoci tento algoritmus slúži na detekovanie rôznych objektov, primárnou motiváciou bola problematika detekcie tváre [1].

Všetky ľudské tváre zdieľajú isté podobnosti. Napríklad, oblasť očí je tmavšia ako oblasť líc. Na detekciu týchto podobností na obraze Viola-Jones využíva Haarové príznaky. Ako trénovací algoritmus sa využíva AdaBoost.

AdaBoost je skratka od plného názvu - Adaptive Boosting. Je to jeden z algoritmov, ktoré sa používajú pri strojovom učení. Vymysleli ho Yoav Freund a Robert Schapire [2], ktorý zaňho v roku 2003 dokonca získali Gödelovú cenu, ktorú udeľuje spoločne *European Association for Theoretical Computer Science* a *Association for Computing Machinery* [3].

Obr. 1.1: Ukážka Haarovového príznaku. Na obrázku môžeme vidieť podobnosť príznaku s ľudskou tvárou. V oblastí očí je tvár tmavšia, ako v oblasti líc. [17]

1.1.2 Rozpoznávanie tvárí

Ak už sme tvár na obraze detekovali, ďalším dôležitým krokom je jej rozpoznanie. Rozpoznávanie tvárí je v dnešnej dobe problematikou, ktorej sa venuje obrovská pozornosť. Aj preto, lebo si nachádza uplatnenie v čoraz väčšej škále odvetví. Za zmienku stoja hlavne tieto:

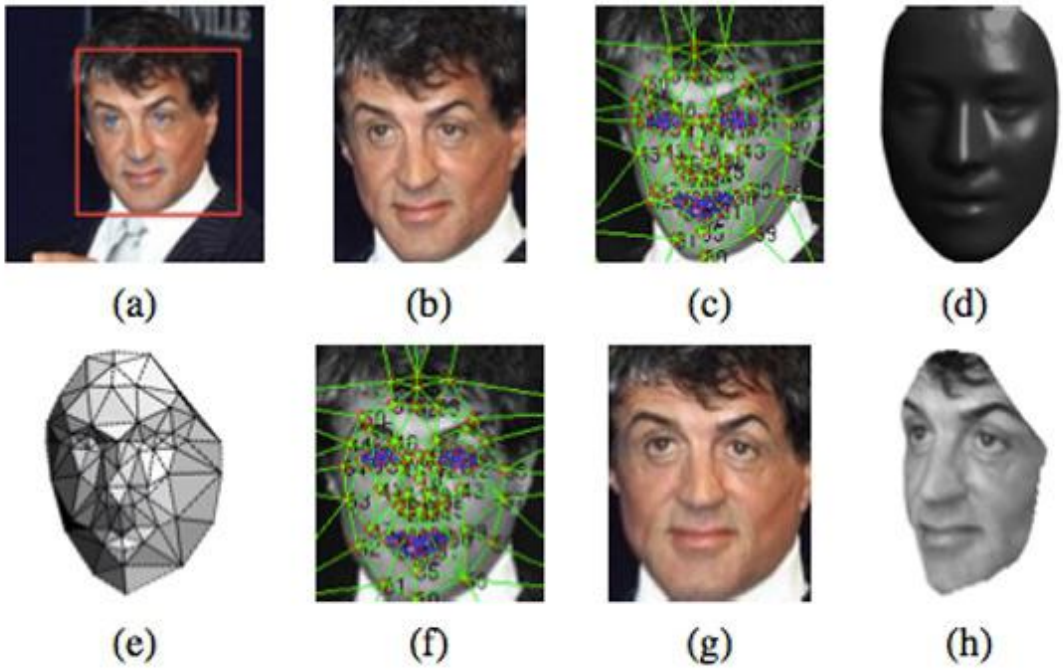
- **Prihlasovanie do systémov -** V dnešnej dobe sme svedkami, že čoraz viac výrobcov mobilných telefónov pristupuje k riešeniu, kedy sa dá telefón odomknúť len rozpoznaním tváre, kedy sa dotyčná osoba pozrie do prednej kamery. Rovnako sa podobné riešenie začínajú integrovať aj do notebookov, kde na snímanie obrazu slúži webkamera. Takéto riešenia v mobilných telefónoch sa používajú aj na prihlasovanie do bankových aplikácií, či dokonca na potvrdenie platby mobilom.

Avšak, tieto systémy rozpoznávania tváre majú aj svoje slabiny. Sú známe prípady, kedy sa tieto systémy dajú jednoducho oklamať fotografiou človeka [4]. Takátu slabinu sa darí eliminovať na minimum za použitia rozličných 3D technológií. Jednou z takých je aj technológia od Apple, Face ID. Tá využíva hĺbkovú mapu tváre, ktorú si vytvára za pomoci viac než 30-tisíc neviditeľných bodov, ktoré sníma infračervená kamera [5].

Obr. 1.2: Technológia Face ID: Na obrázku môžeme vidieť technológiu, ktorú využíva Face ID. Na tvár projektuje viac ako 30-tisíc neviditeľných bodov, ktoré sníma infračervená kamera a tak vytvára hĺbkový 3D model tváre. [18]

- **Sociálne siete -** Dôležitou sférou, v ktorej si rozpoznávanie tvárí našlo svoje uplatnenie, sú aj sociálne siete. Denne naň pribúdajú obrovské množstvá fotografií, na ktorých sa nachádzajú ľudia. A mnoho z týchto ľudí ostáva neoznačených. Preto si spoločnosť Facebook vytvorila vlastnú hlbokú neurónovú sieť, ktorou úlohou je ľudí na fotke rozpoznať a používateľom odporúčať ich označenie. Táto neurónová sieť sa volá **DeepFace** a pozostáva z 9 vrstiev, v ktorých sa dokopy nachádza 120 miliónov prepojení [6]. Navyše, sieť dosahaju pozorudohnú úspešnosť 97.35% [7]. To sa už skoro rovná ľudskej presnosti, ktorá dosahuje úspešnosť 97.53% [6].

  Podobným spôsobom, aj keď s menšou úspešnosťou, fungujú aj rôzne galérie fotografií v telefónoch. Tie vám síce nepovedia, ktora sa na fotke nachádza, no vedia vytvoriť album, v ktorom sa nachádzajú všetky fotografie s dotyčnou osobou. Jej pomenovanie je ale už v réžií používateľa.



Obr. 1.3: Ukážka fungovania neurónovej siete DeepFace. Tá si z pôvodného obrázku (a) vytvorí 3D model tváre (d). Ten vie následne ľubovoľne rotovať a vytvoriť si fotografiu osoby z rozličných uhlov (g, h). [19]

- **Polícia, vojenské účely -** V neposlednom rade netreba zabudnúť ani na policajné, či vojenské účely. Takéto systémy nájdeme hlavne na letiskách, kde sú napojené na policajné databázy s hľadanými osobami. Podobné systémy dnes ale nájdeme aj na všetkých moderných štadiónoch. Návštevník sa pri vstupe musí postaviť pred kameru a odfotiť. Ak systém vyhodnotí, že sa osoba nachádza na zozname ľudí, ktorí majú zakaz vstupu na štadión, organizátor takúto osobu jednoducho nepustí dnu.

  Avšak, ako technológia pokročila, takéto systémy sa zavádzajú aj do oblastí, ktoré sú bližšie k nášmu každodennému životu. Napríklad, metropolitná polícia v Londýne začne používať takéto kamery so systémom rozpoznávania tvárí aj na verejných priestrnstvách. To všetko preto, aby mohla ľahšie odhalovať podozrivé osoby. Objavujú sa aj kritické hlasy, že sa takýmito opatreniami narúšajú občianské slobody, ako napríklad právo na súkromie [8].

1.2 Metódy pre rozpoznávanie tvárí

Nakoľko rozpoznávanie tvárí je široká problematika, existuje viacero metód, ako tvár v obraze rozpoznať. Uvedieme si zopár príkladov, no do podrobna si rozoberieme len

metódy, ktoré nás v tejto práci zaujímajú najviac - Local Binary Pattern Histograms a metódu za pomoci hlbokých neurónových sietí (ResNet) trénovaných pomocou metric loss function.

1.2.1 Eigenfaces

Jednou z mnohých metód je aj metóda nazvaná **Eigenfaces**. Je to metóda, ktorú vytvorili Matthew Turk spolu s Alexom Pentlandom, keď rozšírili prácu Sirovicha a Kirbyho.

Eigenfaces nepopisuje nutne príznaky, ktoré sú pre ľudí intuitívne, ako sú oči, ústa, uši či nos. Namiesto toho sa táto metóda snaží extrahovať relevantné informácie o tvári z množiny obrázkov a zakódovať ich čo najefektívnejšie. Na to mu slúžia eigenvektory. Tieto vektory si môžeme predstaviť ako množinu príznakov, ktoré spolu charakterizujú variáciu medzi obrázkami tváre. Tvár, ktorú potom pomocou eigenvektorov vykreslíme, sa volá eigenface.

Proces rozpoznania tváre prebieha nasledovne [9]:

1. Inicializácia: Získať trénovaciu množinu a vypočítať pre ne eifenfaces, ktoré rezprezentujú priestor tvárí
2. Ak získame nový obrázok tváre, vypočítame preň množinu váh. A to tak, že premietneme obrázok do priestoru tvárí
3. Rozhodneme, či je obrázok skutočne tvár (nezáleží na tom, či ju poznáme alebo nie)
4. Ak ide o tvár, podľa váh z bodu 2 rozhodneme, či ju poznáme alebo nie
5. (Dodatočný bod) Ak sme videli neznámu tvár už viacero krát, zo získaných obrázkov vypočítame eigenface a tvár zaradíme medzi tie, ktoré vieme rozpoznať

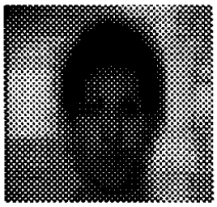
1.2.2 Local Binary Patterns Histogram

Algoritmus Local Binary Patterns Histogram (LBPH) vychádza z algoritmu Local Binary Patterns, ktorý bol po prvýkrát popísaný v roku 1994 [10].

LBPH vznikol ako kombinácia Local Binary Patterns spolu s Histogramom orientovaných gradientov. A hoci sa na prvý pohľad môže zdať LBPH ako jednoduchý, v skutočnosti je aj veľmi efektívny. Algoritmus dosahuje presnosť až 90 percent [11]. Navyše, vie rozpoznať tváre nie len z pohľadu z predu, ale aj zo strán.

A ako vlastne Local Binary Patterns Histogram funguje? Prvým krokom pri LBPH je vypočítať nový obrázok, ktorý lepšie popíše pôvodný obrázok (zvýrazní tvárové charakteristiky). A ako tento nový obrázok získame? Nasledovne [12]:

(a)



(b)

Obr. 1.4: Ukážka trénovacej množiny pre Eigenfaces. Z trénovacej množiny obrázkov tváre (a), vieme vypočítať priemernú tvár popísanú eigenvektormi, tzv. eigenface. [20]

1. Obrázok máme v odtieňoch sivej.

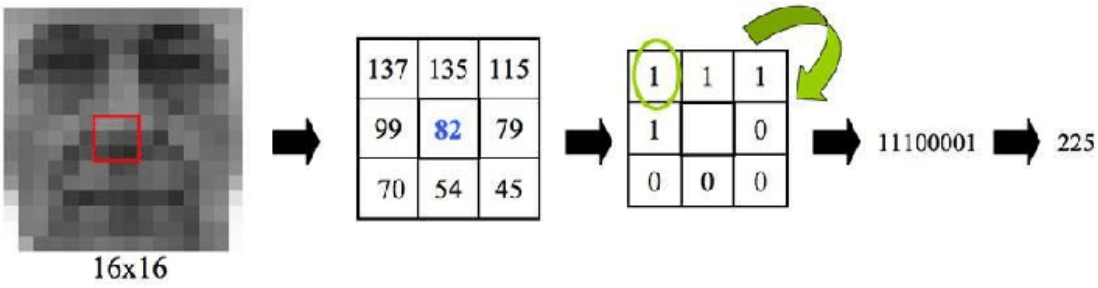
2. Postupne vyberáme "štvorčeky"3x3 pixelov, ktoré reprezentujeme ako maticu 3x3 hodnôt intinzity pixelov (rozmedzie 0 až 255).

3. Následne vyberieme stredný pixel a jeho intenzitu určíme ako prah. Ten nám poslúži na určenie hodnôt zvyšných ôsmich susedov.

4. Pre každého suseda určíme novú hodnotu nasledovne: Ak je hodnota intenzity pixelu vyššia, ako náš prah, susedovi určíme hodnotu 1. Ak nižšia, susedovi priradíme 0.

5. Takto nám vznikne matica, ktorá obsahuje len binárne hodnoty (ignorujeme stred, čiže náš prah). Ak dáme tieto binárne hodnoty do jednej línie, vznikne nám binárne číslo (viď. obrázok 1.5).

6. Toto binárne číslo prevedieme do desiatkovej sústavy a nastavíme ho ako stredovú hodnotu našej matice, tj. hodnotu nastavíme ako novú intenzitu tohto pixelu.

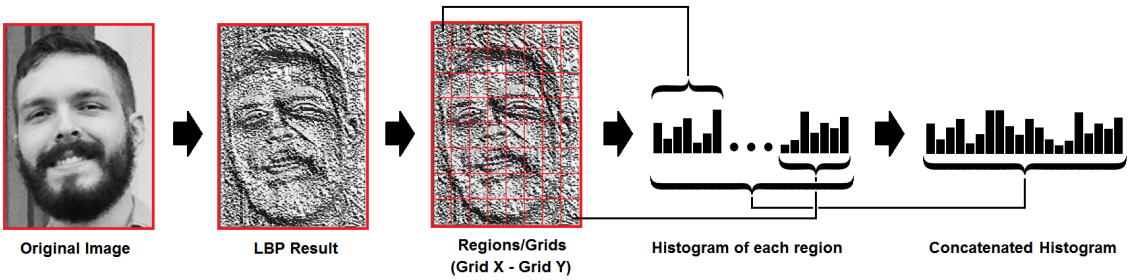


Obr. 1.5: Vyberieme "štvorček"3x3 a určíme si prah, tj. hodnotu intenzity stredného pixelu. Následne určíme binárne číslo a jeho prevedením do desiatkovej sústavy získame novú hodnotu intenzity pre pixel. [21]

Následne potrebujeme z novovzniknutého obrázku získať histogram hodnôt. Ten získame tak, že si obrázok rozdelíme na menšie oblasti a pre každú túto oblasť vypočítane histogram. Keďže obrázok je v odtieňoch sivej, histogram pre každú oblasť bude mať len 256 pozícií (hodnoty v rozmedzií 0 až 255).

Majme náš obrázok rozdelený napríklad na mriežku 8x8 (viď. obrázok 1.6). Náš finálny histogram získame tak, že všetky histogramy pre menšie oblasti spojíme do jedného veľkého. Vo finále tak bude obsahovať 8x8x256 = 16386 pozícií. Tento finálny histogram reprezentuje charakteristiku pôvodného obrázku.

Následne už len treba finálny histogram porovnať s histogramami tvárí, ktoré vieme rozpoznať. To vieme spraviť napríklad tak, že pre každé dva histogramy určíme ich euklidovskú vzdialenosť. Systém nám potom vráti meno tej osoby, ku ktorej je nový histogram najbližšie.



Obr. 1.6: Obrázok si rozdelíme na menšie oblasti a pre každú z nich určíme histogram hodnôt intenzít. Finálny histogram vznikne tak, že všetky histogramy pre jednotlivé oblasti spojíme do jedného veľkého histogramu. [22]

1.3 Použité technológie

V tejto časti si predstavíme jednotlivé technológie, s ktorými budeme pri riešení našej úlohy pracovať. Našim programovacím jazykom bude Python a budeme používať dve jeho knižnice, OpenCV a Dlib.

1.3.1 Python

Python bol vytvorený v roku 1991 Guidom van Rossumom. Je to interpretovaný, vysoko-úrovňový programovací jazyk. Python je dynamicky typovaný a obsahuje garbage collector [13].

Hlavnou filozofiou pri jeho návrhu bola čitateľnosť. To je viditeľné najmä na fakte, že každý ďalší blok kódu musí byť odsadený tabulátorom. K jeho významným vlastnostiam patrí aj to, že je veľmi jednoduchý na učenie.

Python sa momentálne nachádza v dvoch verziách. A to je 2.7.x a 3.5.x. Verzia jazyka 3.0 vyšla ešte v roku 2008 a predstavuje obrovskú revíziu jazyka od jeho základov. Avšak za cenu toho, že tieto dve verzie nie sú plne kompatibilné.

1.3.2 OpenCV

Názov OpenCV predstavuje skratku pre *Open source computer vision*. Tento projekt začal ešte v roku 1999 v spoločnosti Intel. Dnes je však voľne dostupný pod licenciou open-source BSD license.

V súčastnosti je súčasťou tejto knižnice viac ako 2500 optimalizovaných algoritmov, ktoré sa dotýkajú počítačového videnia alebo strojového učenia. Nájdeme tu algoritmy pre detekciu a rozpoznávanie tvárí, objektov, ale aj algoritmy pre sledovanie pohybu očí či na získanie 3D modelu nejakého objektu [15].

OpenCV je písané v jazyku C++, čo zaručuje rýchlosť jednotlivých algoritmov. Avšak, existencia rôznych API a Wrapperov nám umožňuje túto knižnicu používať aj v iných programovacích jazykoch, napríklad aj v Pythone.

1.3.3 Dlib

Dlib je knižnica, ktorej vývoj sa začal ešte v roku 2002. Za ten čas prešla obrovským vývojom a v súčasnosti ponúka široku škálu možností. Nájdeme tu algoritmy a riešenia pre strojové učenie, spracovanie obrazu, threading, networking či GUI a testovanie [16].

Dlib je podobne ako OpenCV, natívne písaná v programovacom jazyku C++.

Literatúra

[1] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.110.4868

[2] https://www.sciencedirect.com/science/article/pii/S002200009791504X?via%3Dihub

[3] http://eatcs.org/index.php/goedel-prize

[4] https://www.mojandroid.sk/smartfony-odomykanie-tvarou-oklamete-fotografiou/

[5] https://support.apple.com/en-us/HT208108

[6] https://www.extremetech.com/extreme/178777-facebooks-facial-recognition-software-is-now-as-accurate-as-the-human-brain-but-what-now

[7] https://research.fb.com/publications/deepface-closing-the-gap-to-human-level-performance-in-face-verification/

[8] https://svet.sme.sk/c/22310001/v-londyne-zacnu-pouzivat-kamery-so-systemom-na-rozpoznavanie-tvari.html

[9] https://sites.cs.ucsb.edu/ mturk/Papers/mturk-CVPR91.pdf

[10] https://ieeexplore.ieee.org/document/576366

[11] https://www.researchgate.net/publication/327980768_LBPH_Based_Improved_Face_Rec

[12] https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b

[13] https://www.python.org

[14] https://opencv.org

[15] https://opencv.org/about/

[16] http://dlib.net/

[17] https://en.wikipedia.org/wiki/File:Haar_Feature_that_looks_similar_to_the_eye_region

[18] https://www.patentlyapple.com/patently-apple/2019/02/an-overview-of-apples-second-round-of-face-id-secrets-published-by-the-us-patent-office-this-week.html

[19] https://slate.com/technology/2014/03/deepface-facebook-face-recognition-software-is-97-percent-accurate.html

[20] https://sites.cs.ucsb.edu/ mturk/Papers/mturk-CVPR91.pdf

[21] https://www.researchgate.net/publication/332130533_Biometric_Attendance_Management_Syster

[22] https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b