FAKULTA MECHATRONIKY,
INFORMATIKY A MEZIOBOROVÝCH
STUDIÍ TUL

Bakalářská práce

Systém pro automatické detekování a identifikaci osob

Studijní program: B0613A140005 Informační technologie
Studijní obor: Inteligentní systémy

Autor práce: **Jan Podávka**
Vedoucí práce: doc. Ing. Josef Chaloupka, Ph.D.
Ústav informačních technologií a elektroniky

Liberec 2023

FAKULTA MECHATRONIKY,
INFORMATIKY A MEZIOBOROVÝCH
STUDIÍ TUL

Zadání bakalářské práce

Systém pro automatické detekování a identifikaci osob

Jméno a příjmení:	**Jan Podávka**
Osobní číslo:	M20000070
Studijní program:	B0613A140005 Informační technologie
Specializace:	Inteligentní systémy
Zadávající katedra:	Ústav informačních technologií a elektroniky
Akademický rok:	2022/2023

Zásady pro vypracování:

1. Seznamte se s problematikou zpracování a rozpoznávání obrazu, s ohledem na řešenou problematiku detekování a identifikování osob z digitálního obrazu.
2. Vytvořte databáze obličejů osob. V databázi by se mělo vyskytovat sto lidí. Každá osoba by měla být v databázi zastoupena alespoň padesáti různými snímky.
3. Otestujte na vytvořené databázi algoritmy pro detekování a identifikaci osob. Zároveň otestujte i algoritmy pro rozpoznávání emocí, pohlaví a věku z digitálního obrazu lidského obličeje
4. Na základě předchozí analýzy (testování), vytvořte program, který bude sloužit pro (polo)automatické detekování osob a jejich identifikaci.

Rozsah grafických prací:
Rozsah pracovní zprávy: 30-40 stran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: Čeština

Seznam odborné literatury:

[1] Šonka, M., Hlaváč V., Boyle. R.: Image processing, analysis, and machine vision. 3rd ed. Toronto: Thomson, 829 s. ISBN 978-0-495-08252-1, 2008
[2] Hlaváč, V., Sedláček, M.: Zpracování signálů a obrazů. 2. přeprac. vyd. Praha: ČVUT, 255 s. ISBN 978-80-01-03110-0, 2007
[3] Liu, Y.: Deep Learning Based Image Processing: Recent Advances and Future Trends. In Eliva Press, 2022. ISBN 978-9994982554.
[4] King, E., D.: Dlib: A Machine Learning Toolkit, In Journal of Machine Learning Research, vol. 10, pp. 1755–1758, 2009.
[5] Geitgey, A.: Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning, In: https://medium.com/, 2016.

Vedoucí práce: doc. Ing. Josef Chaloupka, Ph.D.
Ústav informačních technologií a elektroniky

Datum zadání práce: 24. října 2022
Předpokládaný termín odevzdání: 22. května 2023

L.S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

prof. Ing. Ondřej Novák, CSc.
vedoucí ústavu

V Liberci dne 24. října 2022

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

22. května 2023 Jan Podávka

Systém pro automatické detekování a identifikaci osob

Abstrakt

Tato bakalářská práce se zabývá problematikou detekování obličeje, identifikace osob a rozpoznávání vybraných charakteristických vlastností obličeje (emoce, pohlaví a věk). První část práce je zaměřena na teoretický rozbor metod zvolených na základě rešerše. Praktická část práce se věnuje tvorbě databáze slavných osobností, která následně byla využita pro analýzu metod zvolených v teoretické části. Tato analýza zohledňuje kromě samotné úspěšnosti také možnost nasazení v zařízeních pracujících v reálném čase. Poslední část práce je věnovaná tvorbě multiplatformní aplikace napsané v programovacím jazyce Python za použití knihovny Kivy, sloužící pro tvorbu grafického uživatelského rozhraní. Aplikace nabízí možnosti stažení fotografií, jejich následnou anotaci a vizuální testování metod na libovolném snímku či v reálném čase skrze webkameru.

Klíčová slova: detekce obličeje (segmentace prahováním, Viola-Jones, HOG), identifikace osob (PCA, MACE, DML), rozpoznání charakteristických vlastností obličeje (YOLOv7)

System for automatic detection and identification of people

Abstract

This bachelor thesis deals with the problem of face detection, face recognition and recognition of selected facial features (emotion, gender and age). The first part of the thesis focuses on the theoretical analysis of the methods chosen based on the research. The practical part of the thesis is devoted to the creation of a database of famous personalities, which was then used to analyze the methods chosen in the theoretical part. This analysis takes into account, besides the success rate itself, the possibility of deployment in real-time devices. The last part of the thesis is devoted to the creation of a multiplatform application written in the Python programming language using the Kivy library, used to create a graphical user interface. The application offers the possibility to download photos, annotate them afterwards and visually test the methods on any image or in real time through a webcam.

Keywords: face detection (segmentation by thresholding, Viola-Jones, HOG), face recognition (PCA, MACE, DML), recognition of facial features (YOLOv7)

Poděkování

Rád bych zde poděkoval vedoucímu mé bakalářské práce doc. Ing. Josefovi Chaloupkovi, Ph.D. za cenné rady, podněty a přínosné konzultace.

Obsah

Seznam zkratek

HOG	Histogram orientovaných gradientů
HAAR	Metoda Viola-Jones využívající haarovských filtrů
FPS	Snímků za sekundu
DNN	Hluboká neuronová síť
CNN	Konvoluční neuronová síť
FC	Fully connected
FLD	Facial landmark detection
DML	Deep metric learning
BN	Batch normalizace
MP	Maxpooling vrstva
res	Reziduální blok
conv	Konvoluční vrstva
GPU	Grafická karta
CPU	Procesor

Seznam obrázků

Seznam tabulek

Úvod

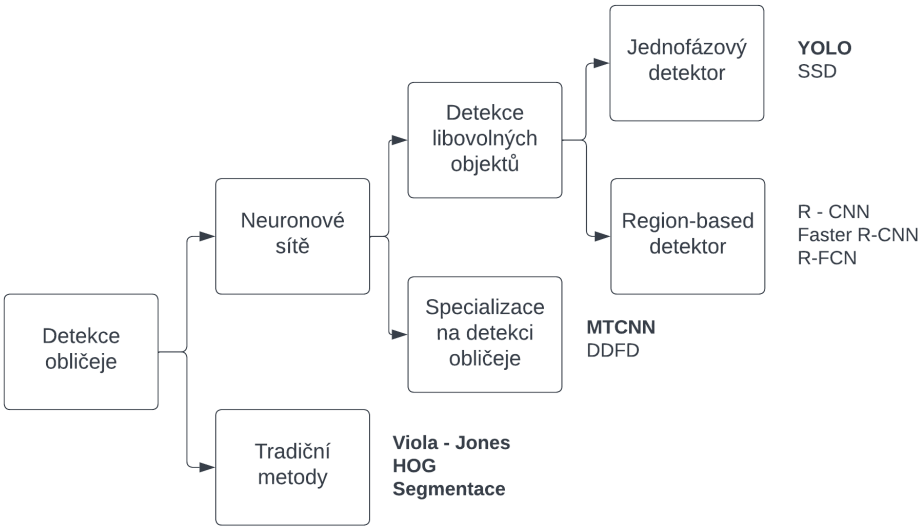
Tato bakalářská práce se zabývá problematikou automatické detekce a identifikace osob z digitálního obrazu. Z počátku byl vývoj zaměřen na statistické modely (PCA, Viola-Jones, HOG), z nichž byly získány příznaky pro následnou analýzu např. metodami strojového učení. Vývoj výpočetní techniky umožnil využití hlubokých neuronových sítích (především konvolučních) se schopností nalézt složitější vzorce mezi daty, které by jinak nemuseli být objeveny. Přestože bývají výpočetně náročnější, vykazují zpravidla vyšší úspěšnost. Avšak kromě úspěšnosti je v některých případech (např. IoT zařízení) potřeba klást důraz také na rychlost, aby bylo možné výsledky zpracovat v reálném čase s minimální latencí. Tato problematika je stále aktuální, využití nachází především v bezpečnosti (např. verifikace osob na letištích či zabezpečovacích systémech domácností). Organizace také mohou nalézt uplatnění v rozpoznávání specifických charakteristik osob, kterými mohou být např. emoce, věk, pohlaví. Tyto znaky mohou dále analyzovat a na jejich základě upravovat svou obchodní strategii.

Cílem této práce je analyzovat jednotlivé metody, jak tradičních statistických modelů, tak modelů konvolučních neuronových sítích, v oblasti detekce obličeje, identifikace osob včetně rozpoznání emocí, věku a pohlaví. K jejich možné analýze byla sestavena a anotována vlastní databáze 100 slavných osobností po 50 fotografiích. V závěrečné části práce je popsána vytvořená multiplatformní aplikace s nástroji umožňujícími sestavení databáze. Dále tato aplikace slouží pro vizuální otestování metod, zvolených v teoretické části práce, s parametry dosahující největší úspěšnosti při testování.

1 Detekce obličeje

Detekce obličeje je počáteční úlohou v procesu identifikace osob či klasifikace jejich charakteristických vlastností (např. pohlaví, věk, emoce). Účel spočívá v nalezení obrazových bodů reprezentující obličej a jejich oddělení od pozadí ohraničujícím boxem (bounding box).

Tuto problematiku lze rozdělit na metody tradiční, založené na využití příznaků získaných z obličeje, a metody využívající neuronových sítí (především konvoluční sítě CNN). Výhodou tradičních metod je menší výpočetní náročnost umožňující nasazení na strojích s menším výkonem, příkladem může být algoritmus Viola-Jones u fotoaparátu mobilního telefonu. Na druhou stranu, neuronové sítě vykazují vysokou úspěšnost i pro náročnější typy snímku (spoustu malých obličejů, různé natočení).

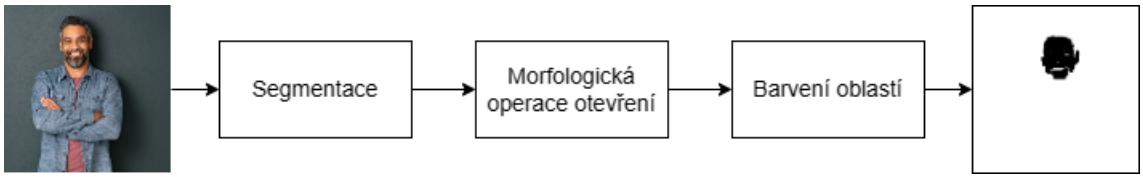


Obrázek 1.1: Diagram dělení detekce obličeje (tučně jsou zvýrazněny využité metody)

1.1 Metoda segmentace prahováním

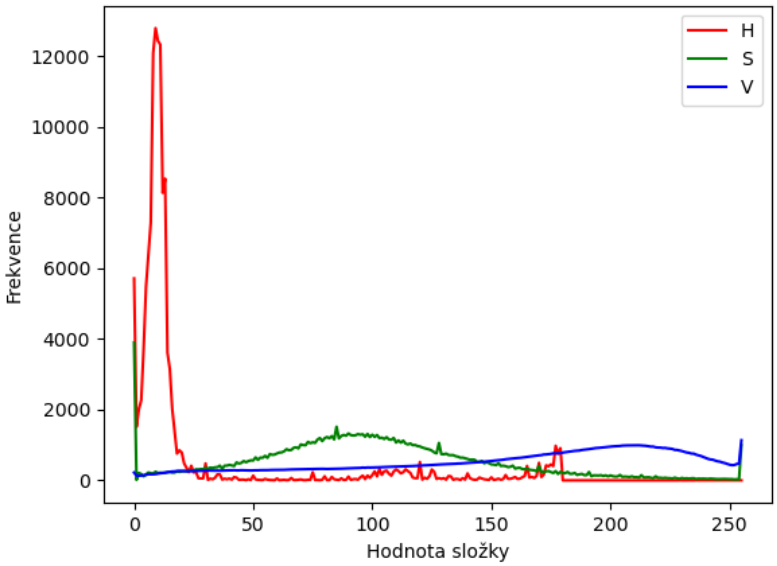
Jedním z nejjednodušších algoritmů používaných k detekci obličeje je segmentace. Tato metoda spočívá v nastavení prahového intervalu hodnot jednotlivých barevných

složek, pro které je pixel klasifikován jako objekt či pozadí. Segmentovaný obraz je možné upravit binární morfologickou operací sloužící k odstranění šumu a zpracovat pro výběr oblasti zájmu (ROI, region of interest).



Obrázek 1.2: Průběh algoritmu segmentace

Před segmentací je vhodné převést vstupní snímek do barevného prostoru HSV, jelikož je méně citlivý na změny vnějšího osvětlení. V této závislosti se především složka Hue (převládající barva) mění relativně méně než např. hodnoty RGB. Zásadním krokem je nastavení prahových hodnot složek HSV. V této práci byla volba těchto hodnot provedena analýzou průměru histogramů z obrazu oblasti obličeje všech osob, nacházejících se v databázi vytvořené v rámci této práce. Z histogramu lze vyvodit vysokou četnost obrazových bodů H složky v úzkém intervalu $h \in [0, 30]$.



Obrázek 1.3: Průměrný histogram databázových snímků

Prahová funkce $f(h, s, v)$ byla vizuální analýzou histogramu určena

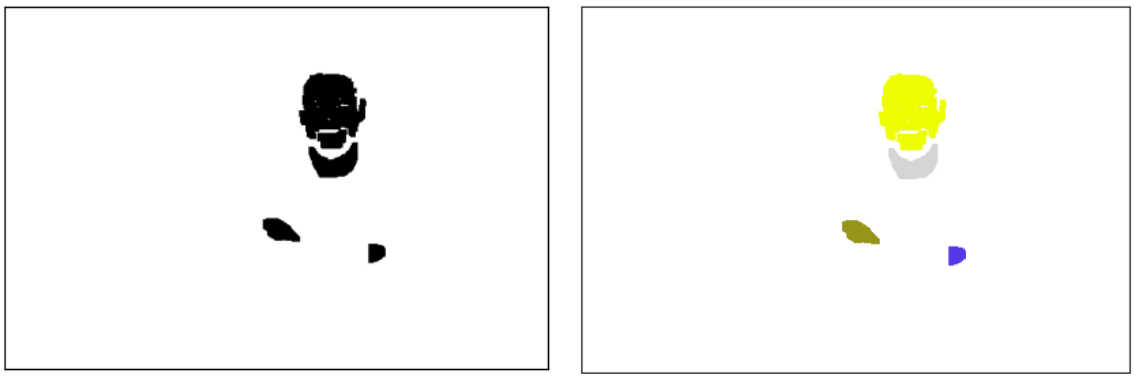
$$f(h,s,v) = \begin{cases} 1 & pokud\, h \in [0,30] \wedge s \in [50,255] \wedge v \in [30,255] \\ 0 & jinak \end{cases} \tag{1.1}$$

(a) původní snímek (b) operace segmentace

Obrázek 1.4: Segmentace

Segmentace na základě barevné informace je náchylná k chybné pozitivní klasifikaci obrazových bodů pozadí. K odstranění binárního šumu byl využit aparát matematické morfologie.[1] Pro účely odstranění binárního šumu byla zvolena operace otevření (eroze následovaná dilatací) s isotropickým strukturním elementem 5×5 (vykazující stejné chování ve všech směrech). Eroze redukuje binární šum a následná dilatace expanduje objekt na původní velikost. V obraze však stále může zůstat i několik jiných objektů, než je obličej. Pro úspěšné pokračování algoritmu musí platit předpoklad, že obličej tvoří největší objekt nacházející se v obraze.

Pro nalezení tohoto největšího spojitého objektu, lze využít algoritmu barvení spojitých komponent (CCL, connected-component labeling). V praktické části této práce byla pro barvení zvolena knihovny Scipy, s využitím výchozí definice spojitosti na 4-okolí. Výstupem algoritmu je označení každého objektu nespojitého s jiným objektem unikátní hodnotou z intervalu $[1, n]$, kde n je počet nespojitých objektů.



(a) morfologická operace otevření (b) barvení oblastí

Obrázek 1.5: Modifikace binárního obrazu

Pro nalezení ROI je zapotřebí výpočtu souřadnic těžiště nejčetnějšího segmentu.

Výpočet těžiště binárního obrazu je

$$x_t = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} x * f(y,x)}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(y,x)} \qquad y_t = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} y * f(y,x)}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(y,x)} \tag{1.2}$$

kde N, M označují výšku a šířku obrazu, x, y jsou hodnoty souřadného systému daného pixelu a $f(y,x) \in \{0,1\}$ reprezentující váhu pixelu.

1.2 Viola-Jones

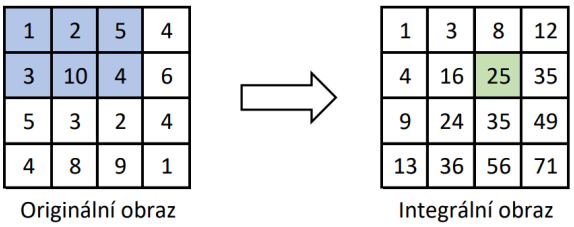
V roce 2001 přišli Paul Viola a Michael Jones s novým algoritmem pro detekování obličeje v obraze, který především díky rychlosti nachází své uplatnění dodnes. Algoritmus Viola-Jones využívá k určení oblasti zájmu haarovských příznaků (haar-like feature). Tyto příznaky jsou založeny na předpokladu, že jasový rozdíl vybraných přechodů obličeje (např. špička nosu–strany nosu) je pro každý obličej podobný.

Obrázek 1.6: Obrazové příznaky (převzato z [2])

Příznak je vypočten jako suma hodnot pixelů bílé části příznaku, odečtená od sumy hodnot pixelů černých. Používají se příznaky složené z 2, 3 a 4 obdélníků. Výpočet těchto příznaků škálovaných na různé velikosti posuvným okénkem přes původní snímek by bylo výpočetně velmi náročné. Autoři proto zveřejnili tři nové klíčové prvky, které vedly k výraznému zvýšení rychlosti detekce. Jedná se konkrétně o integrální obraz, modifikaci algoritmu AdaBoost a kaskádu klasifikátorů.

Integrální obraz umožňuje efektivní způsob součtu hodnot obdélníkové oblasti obrazu v konstantním čase. Výpočet integrálního obrazu $ii(y,x)$ je dán součtem jasových hodnot nacházejících se v původním obraze $f(y,x)$ nad a vlevo od něj.

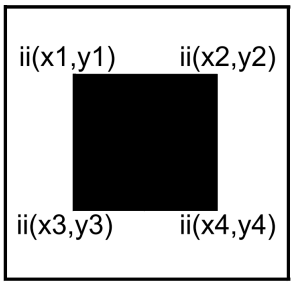
$$ii(y,x) = \sum_{i=0}^{i\leq x}\sum_{j=0}^{j\leq y} f(j,i) \tag{1.3}$$



Obrázek 1.7: Integrální obraz

V originálním obraze by se jeden obdélník příznaku musel spočítat průchodem každého obrazového bodu, integrální obraz umožňuje výpočet pouze ze 4 hodnot. Výpočet součtu obdélníkové oblasti D vypadá následovně

$$D = ii(x_4, y_4) - ii(x_3, y_3) - ii(x_2, y_2) + ii(x_1, y_1) \tag{1.4}$$



Obrázek 1.8: Výpočet obdélníku

Přestože integrální obraz umožňuje efektivní způsob výpočtu příznaků, je časově náročné pro každé posuvné okénko provést kompletní sadu příznaků. Z tohoto důvodu se autoři rozhodli využít trénovacího algoritmu AdaBoost (adaptive boosting). Myšlenkou je, že kombinací několika slabých klasifikátorů (v případě algoritmu Viola-Jones haarovských příznaků) lze vytvořit jeden klasifikátor silný. Vstupem pro trénování algoritmem AdaBoost jsou pozitivní i negativní snímky a počet iterací T (počet vybíraných slabých klasifikátorů). Váhy jsou inicializovány na $w_1 = \frac{1}{2m}, \frac{1}{2l}$, kde m, l je počet negativních, respektive pozitivních snímků. Hlavní cyklus algoritmu provádí T iterací, v každé iteraci je provedena normalizace vah w

$$w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{T} w_{t,j}} \tag{1.5}$$

kde t je aktuální iterace a i je index vstupního data. Následně je vybrán 1 slabý klasifikátor s nejmenší chybou

$$e_t = \sum_{i=0}^{N} w_i |f_j(x_i) - y_i| \tag{1.6}$$

kde N je počet vstupních dat x, j index slabého klasifikátoru, $y \in \{0, 1\}$ symbolizuje skutečnou třídu a e označuje chybu. Poté jsou aktualizovány váhy pro další iteraci

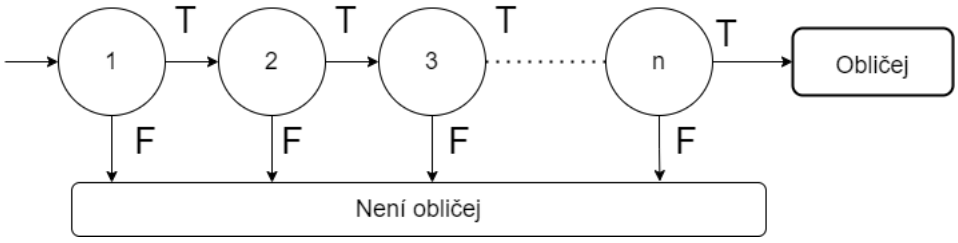
$$w_{t+1,i} = w_{t,i} \beta_t^{1-|f(x_i)-y_i|} \tag{1.7}$$

kde $\beta_i = \frac{e_t}{1-e_t}$ a $f(x)$ je slabý klasifikátor s nejmenší chybou. Výsledný silný klasifikátor je dán lineární kombinací klasifikátorů slabých

$$F(x) = \begin{cases} 1 & pokud \ \sum_{t=1}^{T} \alpha_t f_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & jinak \end{cases} \tag{1.8}$$

kde $F(x)$ je silný klasifikátor, $f_n(x)$ klasifikátor slabý a $\alpha_t = \frac{1}{\beta_t}$ [3].

Výrazné zrychlení detekce přineslo zavedení tzv. kaskády klasifikátorů. Ta se skládá z 1 až n boostovaných (silných) klasifikátorů seřazených v kaskádě za sebou. V prvních stupních jsou využity menší klasifikátory, které mají zamítnout co největší počet negativních okének, ale zároveň detekovat téměř všechny pozitivní. Tím je docíleno, že většina okének je zamítnuta v prvních stupních a není třeba výpočtu ostatních příznaků. V každém dalším stupni kaskády jsou pak více komplexní klasifikátory. Tyto požadavky jsou splněny při trénování AdaBoostu, u kterého je dle potřeby zvyšována prahová hodnota, čímž jsou minimalizovány počty falešně pozitivních klasifikací. Okénko je klasifikováno jako pozitivní v případě, kdy není zamítnuto žádným stupněm kaskády [4].



Obrázek 1.9: Kaskádní klasifikátor

1.3 Histogram orientovaných gradientů

Myšlenkou metody HOG (Histogram orientovaných gradientů) je popis objektu skrze rozložení lokálních intenzit a směrů gradientu. Gradient spojité funkce n proměnných je definován jako

$$\nabla f(x_1, x_2, \ldots, x_n) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right) \tag{1.9}$$

Výkonost detektoru je citlivá na způsob, jakým je gradient vypočten. Derivaci obrazu lze aproximovat konvolucí. Analýza, provedená autory metody HOG poukazuje, že nejúspěšnější je využití dvou konvolučních masek aproximující derivaci se směru osy x a y

$$g_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \qquad g_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \tag{1.10}$$

Pro diskrétní obrazovou funkci ($n = 2$) se gradient skládá z velikosti (modulu) [5]

$$|G| = \sqrt{g_x^2 + g_y^2} \tag{1.11}$$

a směru

$$\theta = \arctan \frac{g_y}{g_x} \tag{1.12}$$

(a) původní snímek (b) modul gradientu (c) směr gradientu

Obrázek 1.10: Zobrazení složek gradientu

Vstupní obraz je rozdělen na menší buňky o velikosti $n \times n$. Pixel je reprezentován hodnotou velikosti $|G|$ a směru gradientu θ, čímž jsou rozměry buňky rozšířeny na $n \times n \times 2$. Pro každou buňku je spočten lokální histogram orientace gradientů. Tento histogram je možné sestrojit dvěma způsoby, první možností je využití celé škály $0° - 360°$. V případech, kdy není kladen takový důraz na směr, je možné využití pouze kladných úhlů $0° - 180°$, čímž je zredukována dimenze příznaků. Autoři v původní dokumentaci uvádějí, že nejlepších výsledků dosahovali rozdělením histogramu na 9 tříd rovnoměrně rozložených v intervalu $0° - 180°$, přidáním počtu tříd docházelo pouze k nepatrnému navýšení úspěšnosti a při využití záporného směru gradientu úspěšnost dokonce poklesla. Lokální normalizace pro zajištění robustnosti vůči prudkým jasovým změnám je provedena sloučením $k \times k$ buněk do jedné větší (tzv. bloku) a provedením L_1 či L_2 normalizace. Finální klasifikace je prováděna na sloučeném vektoru všech normalizovaných bloků nejčastěji klasifikátorem SVM [6].

1.4 MTCNN

MTCNN (Multi-task Cascaded CNN) je systém kaskádově zapojených CNN pro úlohy detekce obličeje a nalezení orientačních bodů obličeje (facial landmark detection). Jedná o nalezení 5 bodů - levé oko, pravé oko, levý koutek úst, pravý koutek úst a špička nosu.

1.4.1 Proces trénování

K trénování jsou využity tři úlohy:

Klasifikace obličeje

- Jedná se klasifikaci, zda se na snímku nachází/nenachází obličej. Jelikož se jedná o binární problém, byla jako loss funkce zvolena binární křížívá entropie (cross-entropy)

$$L^{det} = -(y^{det} \log(p) + (1 - y^{det}) \log(1 - p)) \tag{1.13}$$

  kde y^{det} symbolizuje skutečnou třídu (obličej/není obličej) a p je pravděpodobnost na výstupu sítě z aktivační funkce sigmoida.

Regrese ohraničujícího boxu

- Pro každé kandidátní okno (candidate window) je spočten euklidovský loss

$$L^{box} = \left\| \hat{y}^{box} - y^{box} \right\|_2^2 \tag{1.14}$$

  kde $\hat{y}^{box} \in \mathbb{R}^4$ je predikovaný box, definovaný souřadnicemi x_1,y_1 (lokalizující levý horní roh), šířkou w a výškou h. Skutečné ohraničení boxu je označeno y^{box}.

Lokalizace orientačních bodů obličeje

- Tato úloha je obdobně regresním problémem, proto výpočet hodnoty loss je totožný

$$L^{landmark} = \left\| \hat{y}^{landmark} - y^{landmark} \right\|_2^2 \tag{1.15}$$

  kde odhadované souřadnice všech 5 orientačních bodů (tzv. landmarků) jsou označeny $\hat{y}^{landmark} \in \mathbb{R}^{10}$ a $y^{landmark}$ symbolizuje souřadnice skutečné.

Celkový loss je vypočten jako suma přes jednotlivé loss funkce

$$L = min \sum_{i=1}^{N} \sum_{j \in skut,box,landmark} \alpha_j \beta_i^j L_i^j \tag{1.16}$$

Parametr α označuje váhu (důležitost), která se nastavuje v jednotlivých fázích, $\beta \in \{0, 1\}$ typ snímku a N počet trénovacích snímků [7].

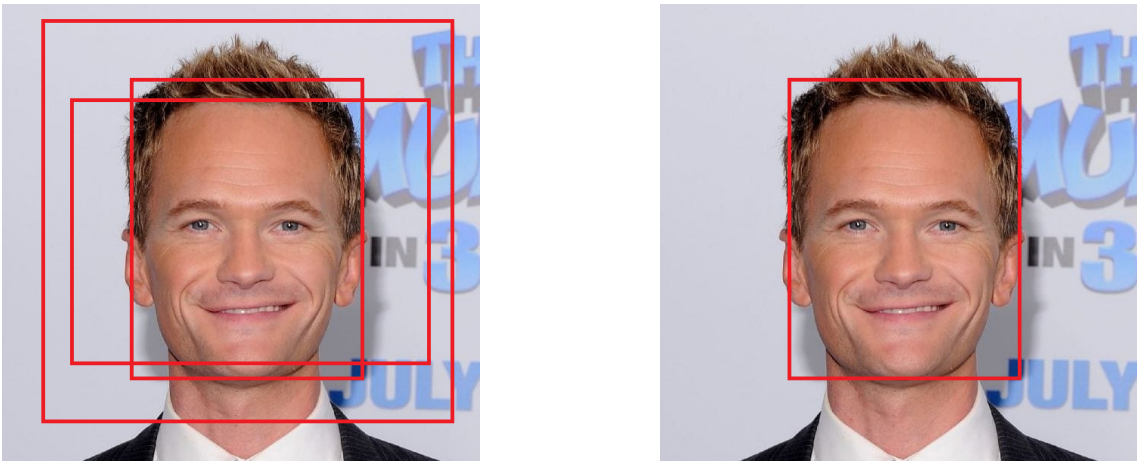
1.4.2 Non-maximum suppression

Algoritmus non-maximum suppression (dále NMS) slouží pro potlačení překrývajících se predikovaných oblastí, které detekují stejný objekt. Jeho průběh je následující [8]:

- sestupné setřídění všech predikovaných boxů dle skóre (confidence)
- výběr boxu s největší hodnotou skóre
- výpočet IoU zvoleného boxu se všemi ostatními překrývajícími

- odstranění boxů, které dosahují hodnoty IoU větší než nastavená hodnota prahové hodnoty
- výběr dalšího boxu a opakování předchozích tří kroků algoritmu, dokud zbývají predikované boxy

Tímto je zredukován počet překrývajících se boxů na ten s největší váhou, je však důležité vhodně zvolit parametr prahové hodnoty, jelikož při objektech nacházejících se blízko sebe může dojít k potlačení správně rozeznané detekce.



(a) predikované boxy (b) aplikace NMS

Obrázek 1.11: Non-maximum suppression

1.4.3 Architektura CNN

Vstupním snímkům je prvně několikrát změněna velikost, až tvoří pyramidu, která je vstupem tří kaskádově zapojených CNN [7]:

P-Net

- V první sítí P-net (Proposal network) dochází k nalezení kandidátních oken a regresi vektoru jejich ohraničujících boxů. Na kandidátní okna je následně aplikován algoritmus NMS.

R-Net

- Každé kandidátní okno z první sítě je vstupem do R-Net (Refine network). V této síti dochází ke zpřesnění ohraničujících boxů a k zamítnutí velké části FP. Následuje opětovné aplikování algoritmu NMS. První dvě sítě mají shodně nastavený parametr $\alpha_{det} = 1, \alpha_{box} = 0,5, \alpha_{landmark} = 0,5$.

O-Net

- Poslední síť O-Net je velmi podobná té předchozí R-Net, ale více se zaměřuje na nalezení 5 orientačních bodů. Toho je docíleno zvýšením váhového parametru $\alpha_{landmark} = 1$.

2 Identifikace osob

Úlohou rozpoznávání osob je určit, která osoba se na daném snímku nachází. Problematiku lze rozdělit na verifikaci (1 ku 1), tedy porovnání dvou snímků a rozhodnutí, zda se jedná o osoby stejné či nikoliv. Druhou možností je identifikace (1 ku n), při které je snímek přiřazen k jedné z n možných osob. Rozpoznávání osob předchází detekce obličeje, v této práci je pro identifikaci osob výchozím detektorem metoda HOG.

2.1 Analýza hlavních komponent

Pro snížení velikosti korelovaných vícerozměrných dat lze využít transformaci nazývanou analýza hlavních komponent (Principal Component Analysis, PCA). Transformací dat do vlastního prostoru (eigenspace) jsou získány již nekorelované příznaky. Tyto příznaky o zredukované dimenzi přesto zachycují většinovou část rozptylu vstupních dat. Vstupní snímky x_i o stejné velikosti jsou převedeny na sloupcové vektory. Seřazením vektorů je vytvořena matice dat $W_x = [x_1, x_2, \dots x_n]$. Pro normalizaci případných velkých jasových rozdílů mezi jednotlivými snímky, je odečten průměrný vektor x_p matice W_x získán z průměru řádků

$$W_i = W_{x_i} - x_p \tag{2.1}$$

kde W je normalizovaná matice dat. Rozptyly normalizovaných dat jsou získány z diagonály kovarianční matice C, vypočtené

$$C = W^T W \tag{2.2}$$

Vlastní čísla λ (a korespondující vlastní vektory) kovarianční matice udávají podíl celkového rozptylu $R \in \{0, 1\}$, jehož hodnotu lze pro k-té největší vlastní číslo spočítat jako

$$R_k = \frac{\lambda_k}{\sum_{i=0}^{N} \lambda_i} \tag{2.3}$$

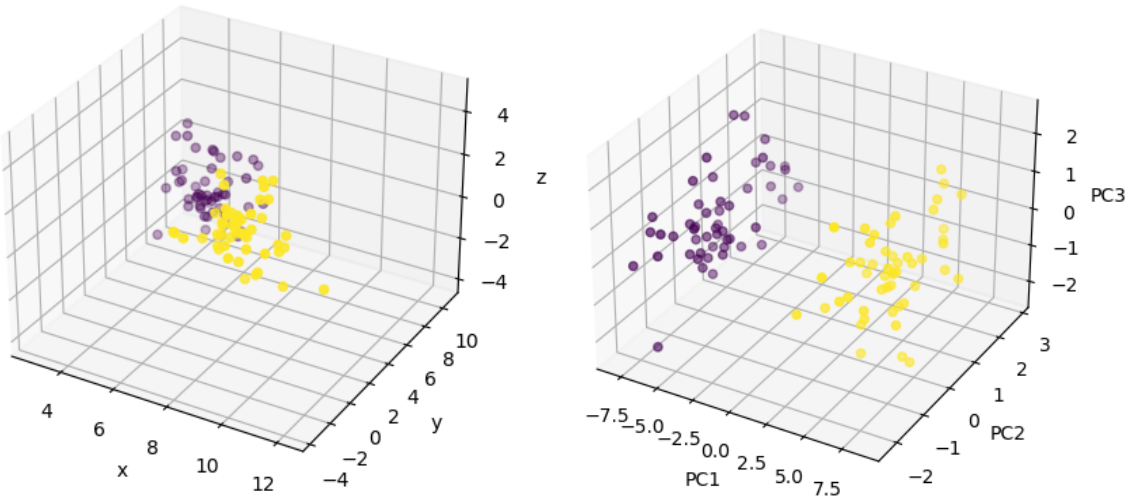
kde N je celkový počet hlavních komponent. Pokud je redukce provedena využitím prvních n hlavních komponent, tedy těch zachycující největší podíl variability.

$$R_n = \frac{\sum_{i=0}^{n} \lambda_i}{\sum_{j=0}^{N} \lambda_j} \tag{2.4}$$

Seřazením n vlastních vektorů sestupně, dle velikosti vlastního čísla, je vytvořena matice E_P. Data mohou být následně promítnuta do nového vlastního prostoru E, zachycujícího poměrnou část celkového rozptylu.

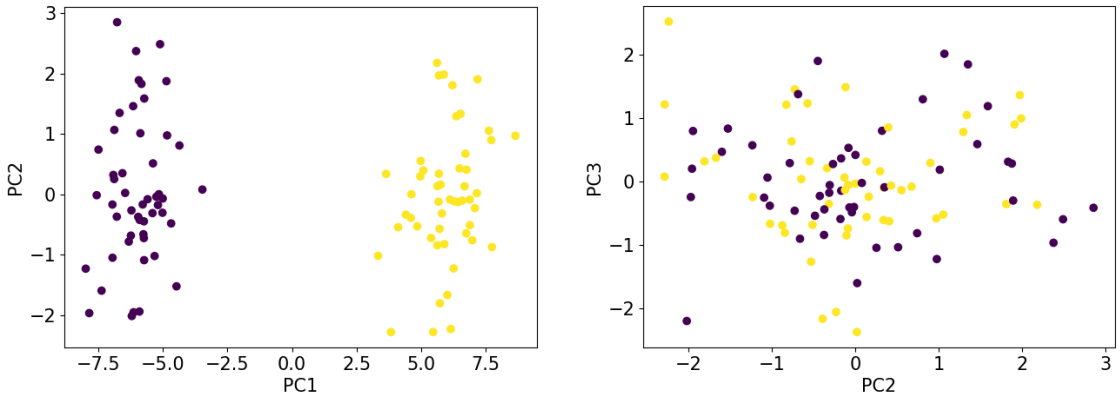
$$E = WE_p \tag{2.5}$$

Pro data promítnutá do vlastního prostoru (viz. obrázek 2.1) je poměr $R_{PC1,PC2,PC3} = (0.95, 0.03, 0.02)$. Z vyobrazených grafů je viditelné, že první hlavní komponenta obsahuje většinovou část celkového rozptylu. To lze znázornit vykreslením jednotlivých komponent jednotlivých hlavních komponent. [9] V úloze



(a) původní prostor (b) vlastní prostor

Obrázek 2.1: Projekce do vlastního prostoru



(a) znázornění PC1 a PC2 (b) znázornění PC2 a PC3

Obrázek 2.2: Vykreslení jednotlivých hlavních komponent

identifikace obličeje je volbou vlastních vektorů (eigenface) definován vlastní prostor (facespace), do kterého jsou promítnuta testovací data a následně porovnána

s promítnutou trénovací sadou na základě metody nejbližšího souseda. V této práci byly využity vzdálenostní metriky Manhattanská vzdálenost L_1

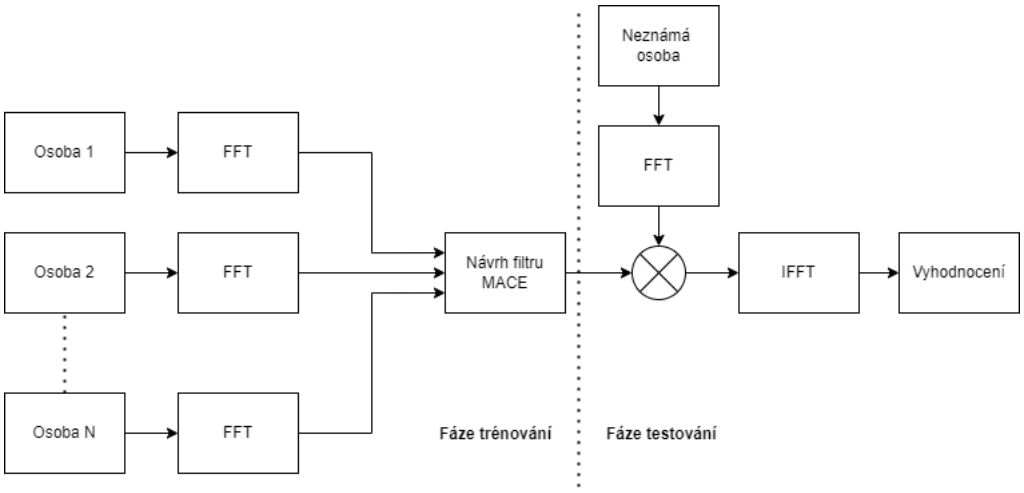
$$L_1(x, y) = \sum_{i=0}^{n} |x_i - y_i| \tag{2.6}$$

a Euklidovská vzdálenost L_2

$$L_2(x, y) = \sum_{i=0}^{n} \sqrt{(x_i - y_i)^2} \tag{2.7}$$

2.2 Korelační filtr Mace

Princip MACE (Minimum Average Correlation Energy) spočívá v minimalizaci průměrné korelační energie. To má za následek, že výsledná korelační rovina je blízká 0 všude, kromě počátku. Identifikace je provedena křížovou korelací mezi navrženým filtrem MACE z trénovacích dat s neznámým obrázkem a testovaným snímkem. Identifikace osoby na testovacím snímku je provedena na základě výsledné korelační roviny.

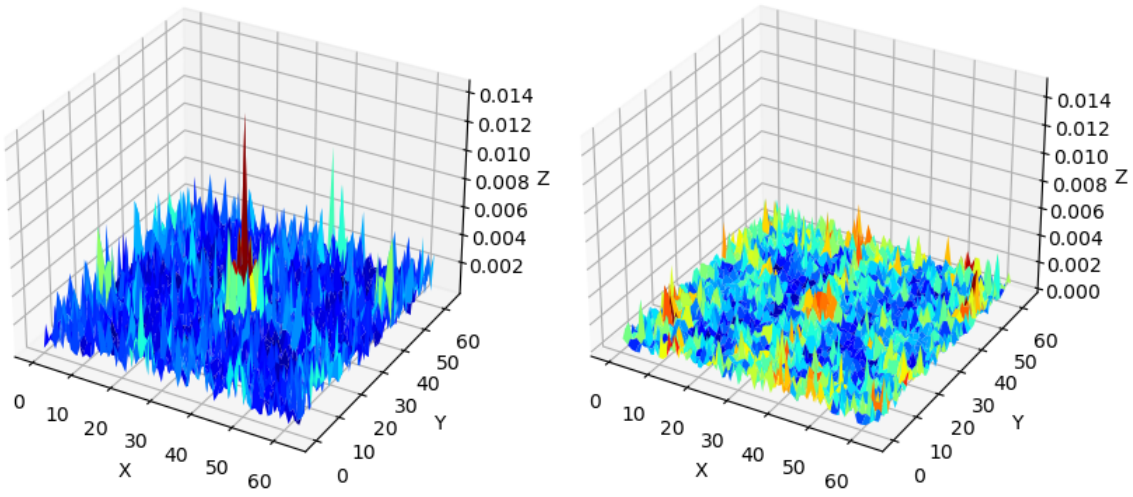


Obrázek 2.3: Diagram průběhu trénování/testování filtru Mace

Průběh učení spočívá v navržení korelačního filtru MACE pro každou třídu nacházející se v trénovací sadě. Vstupní trénovací obrázky dané třídy o počtu N s rozměry $x \times x$ jsou převedeny do šedotónového barevného prostoru a transformovány na sloupcový vektor o velikosti d. Je provedena dvourozměrná rychlá Fourierova transformace (FFT) a lexikografické řazení. Z výsledných vektorů x_{fft} je vytvořena matice X o velikosti $d \times N$. Diagonální matice D o dimenzi rovné $d \times d$ obsahuje na diagonále hodnoty spočtené jako průměr z výkonového spektra řádků matice X. Matice transponovaná a komplexně sdružená k X je označena X^+. Vektor u s N elementy obsahuje předem specifikované korelační vrcholy. Dále budou předpokládány

hodnoty elementů vektoru u váhy 1. Výsledný korelační filtr h_{MACE}, daný sloupcovým vektorem o rozměru d, je přerozdělen na původní rozměr obrázků trénovací sady [10].

$$h_{MACE} = D^{-1}X(X^{+}D^{-1}X)^{-1}c \tag{2.8}$$



(a) pozitivně rozpoznaná třída (b) negativně rozpoznaná třída

Obrázek 2.4: Výsledná korelační rovina

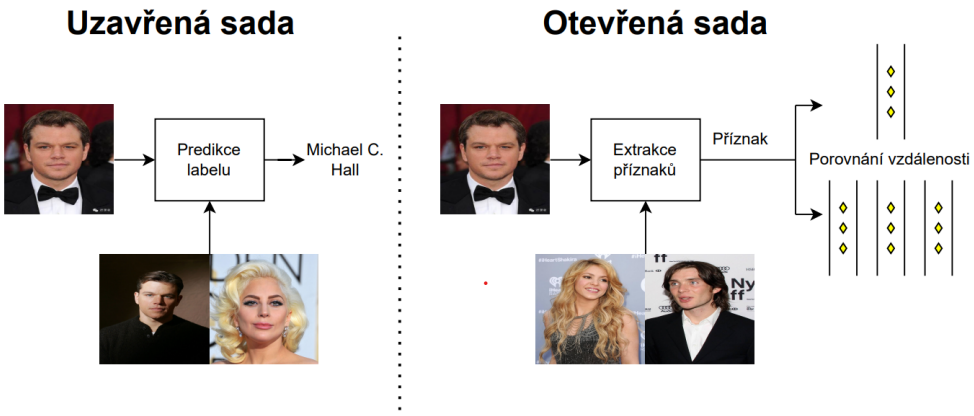
Vyhodnocení bývá prováděno posouzením ostrosti vrcholu PSR (Peak to sidelobe ratio) definovaném

$$PSR = \frac{p - \mu}{\sigma} \tag{2.9}$$

kde p je maximální hodnota vrcholu, μ průměr okolí a σ standardní odchylka okolí. Okolí pro snímek o velikosti 64×64 je obvykle voleno o velikosti 25×25, avšak není zde zahrnut samotný vrchol o rozměrech 5×5. Klasifikace je provedena nalezením nejostřejšího vrcholu odpovídajícího nejvyšší hodnotě parametru PSR.

2.3 Deep metric learning

Situace, při které jsou k dispozici všechny testované osoby v trénovací sadě, není v reálném světě vždy možná. Proto lze tento problém rozpoznávání obličeje rozdělit na tzv. uzavřené sady (closed-set) a otevřené sady (open-set). Pro uzavřené sady platí, že testované osobnosti se objevují v trénovací sadě, zatímco pro otevřené sady se objevovat nesmějí. To se týká úlohy identifikace osoby i verifikace. Cílem Deep metric learning (dále DML) je naučit se vzdálenostní (podobnostní) funkci pomocí hlubokých neuronových sítí tak, aby vstupní vektory x_1, x_2 stejné třídy byly na výstupu neuronové sítě v prostoru co nejblíže u sebe. Naopak vektory patřící do různé třídy byly od sebe co nejvzdálenější. Výsledné příznaky y_1, y_2 je pak mezi sebou možné porovnávat klasickými vzdálenostními metrikami [11].



Obrázek 2.5: Rozdělení úlohy rozpoznávání obličeje dle typu sady

Pro přehlednou implementaci byla v této práci zvolena knihovna `face_recognition`, která je postavena na sadě nástrojů (toolkit) dlib. Přestože je dlib napsán v jazyce C++, disponuje rozhraním umožňující využívat nástroje i v jazyce Python. Z důvodu využití právě těchto nástrojů, bude následující zbytek kapitoly zaměřen na technologie, které jsou v dlib využity.

2.3.1 Architektura sítě

Autoři využili modifikovanou síť ResNet-34 z článku Deep Residual Learning for Image Recognition od He, Zhang, Rena a Suna [12] s 28 konvolučními vrstvami a počtem filtrů sníženém na polovinu. Síť byla natrénována na milionech snímcích s celkem 7485 unikátními osobnostmi. Výstupem sítě je vektor o délce 128, který určuje daný obličej. Pro účely DML nelze využít klasický softmax loss, který přestože dokáže oddělit příznaky, tak je dostatečně nediskriminuje. Z toho důvodu byly vymyšleny loss funkce, které mají mnohem větší diskriminační sílu.

2.3.2 Triplet loss

Přestože se jedná o loss funkci, která již není považována za nejúspěšnější (překonána např. ArcFace, SphereFace či CosFace), tak je stále řazena mezi velmi úspěšné. Implementace Dlib využívá právě této loss funkce.

Pro výpočet triplet loss je třeba tří typů snímků, z nichž první je tzv. anchor X_A. Každý anchor pak vyžaduje jeden pozitivní snímek X_P (tedy snímek stejné třídy) a jeden snímek negativní X_N (snímek jiné třídy). Výpočet triplet loss pro jednu trojici snímku je dán rovnicí

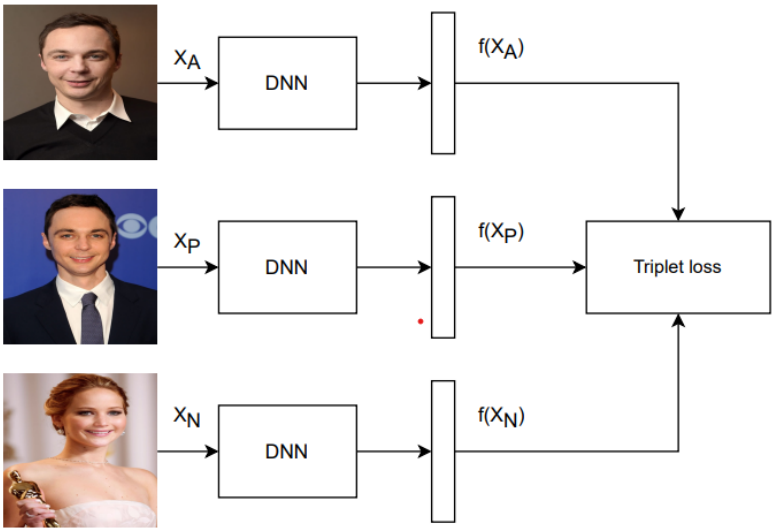
$$L = \max(0, d^+ - d^- + \alpha) \tag{2.10}$$

kde

$$d^+ = \|f(x_A) - f(x_P)\|_2^2 \tag{2.11}$$

$$d^- = \|f(x_A) - f(x_N)\|_2^2 \tag{2.12}$$

Pokud vzdálenost d^+ bude větší $d^- + \alpha$, tak $L > 0$, v opačném případě $L = 0$. Dochází k minimalizování vzdálenosti snímků stejné třídy od snímku s různou třídou [13].



Obrázek 2.6: Triplet loss

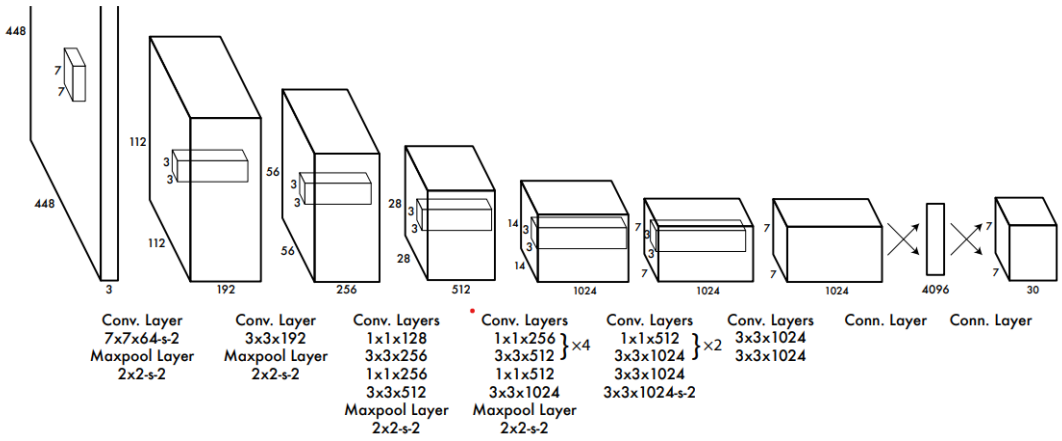
3 Rozpoznávání pohlaví, věku a emocí

Kromě samotné identifikace osob je možné také rozpoznávat různé charakteristické vlastnosti obličeje. V této práci jimi jsou pohlaví, emoce a věk. Vybrány byly dva způsoby, oba využívající konvolučních neuronových sítí. První z nich byla Yolov7, umožňující rychlé detekování objektů a současnou klasifikaci jednotlivých tříd. Druhým způsobem bylo vytvoření programu sloužícího pro navržení vlastní struktury neuronových sítí. Nevýhodou tohoto způsobu je nutnost připojení detektoru pro nalezení oblasti obličeje.

3.1 YOLO

V roce 2016 byl představen algoritmus YOLO (You Only Look Once), řadící se mezi jednostupňové detektory umožňující, na rozdíl od Region-based detektorů, nalézt objekty v obraze pouze v jednom průchodu. Kromě samotné detekce je provedena i klasifikace tříd. Absence více průchodů umožnila detekování objektů v reálném čase s minimální latencí. Přestože již byla uvedena osmá verze YOLO, v této práci byla využita YOLO verze 7, jelikož v době tvorby práce byla nejaktuálnější.

Původní síť se skládá z 24 konvolučních vrstev, následovaných 2 plně propojenými vrstvami (FC vrstva).



Obrázek 3.1: Architektura sítě (převzato z [14])

Vstupní obraz o velikosti $n \times n$ je rozdělen na mřížku o velikosti buněk $S \times S$. V každé buňce je predikováno B ohraničujících boxů, přičemž každý je složen

z x, y, w, h a confidence (důvěryhodnosti), kde x, y označují souřadnice středu relativně vzhledem k buňce,w šířku, h výšku a c označuje míru důvěry modelu v danou predikci. Výpočet confidence je

$$c = Pr(objekt) * IOU \tag{3.1}$$

Dále je v každé buňce spočtena podmíněná pravděpodobnost C tříd $PR(C_i|objekt)$ (pouze jednou pro každou třídu, nezávisle na počtu B), tím je vytvořena pravděpodobností mapa pro celý snímek a lze k ní přistupovat jako k regresnímu problému. Při detekci je poté vynásobena confidence s podmíněnou pravděpodobností, čímž je získána hodnota, reprezentující váhu, jaká je přikládána tomu, že se daný objekt v buňce opravdu nachází. Výstupní vektor jednoho snímku má rozměry

$$S \times S \times (B \cdot 5 + C) \tag{3.2}$$

Tento vektor zajišťuje, že v každé buňce lze při detekování spočítat, který objekt se v ní nachází a s jakou pravděpodobností. Zároveň zajišťuje, že není zapotřebí druhého průchodu. Tím ale může vzniknout spoustu ohraničujících boxů, proto je na výstup aplikován algoritmus NMS (viz. kapitola 1.4.2) [14].

4 Vyhodnocení použitých metod

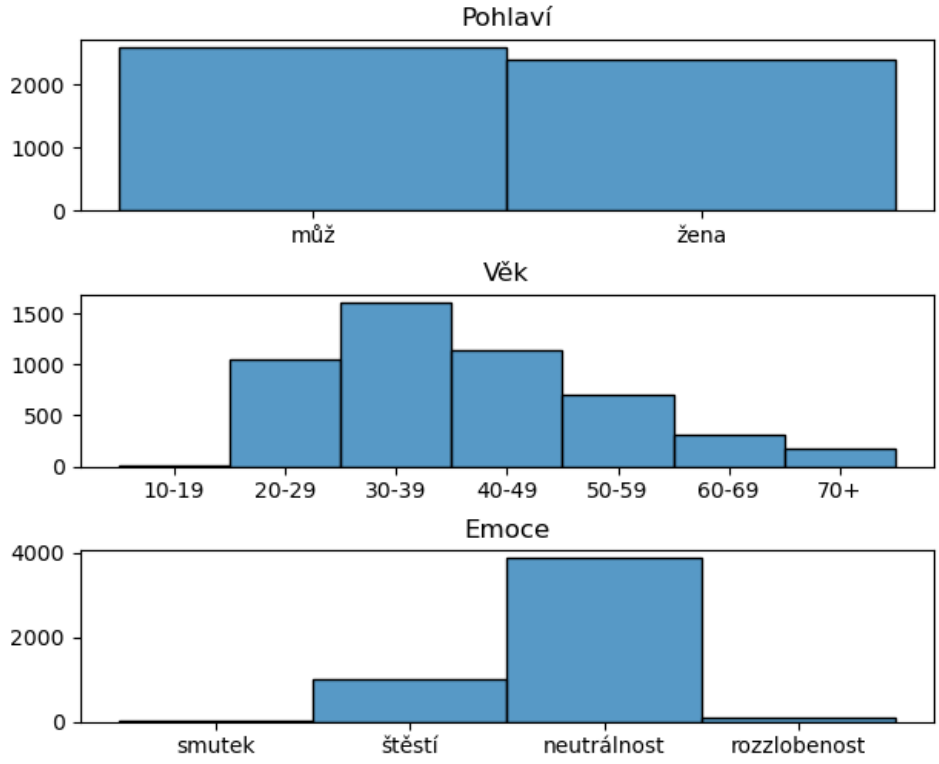
4.1 Databáze

V rámci bakalářské práce byla vytvořena databáze čítající celkem 100 slavných osobností. Každá osoba je zastoupena 50 různými snímky, přičemž byl kladen důraz na největší možnou variabilitu (odlišné světelné podmínky, kvalita obrazu, časový odstup). Ohraničení obrazové oblasti obličeje probíhalo poloautomatickou formou s využitím detekčního algoritmu HOG. Zbylé kategorie byly stanoveny prostřednictvím vytvořené aplikace. Pro uložení anotací k snímkům byl zvolen formát JSON s výchozím umístěním v projektu */data/data.json* a následující strukturou:

- **"name"**: jméno
- **"age"**: 10 | 20 | 30 | 40 | 50 | 60 | 70
- **"gender"**: "F" | "M"
- **"emotion"**: "S" | "H" | "N" | "A" |
- **"faceloc"** = $[x_1, y_1, y_2, x_2]$
- **"path"** = "Osobnost/jméno osobnosti/název snímku"

Kategorie věku byla rozdělena na 7 tříd, rovnoměrně rozložených po 10 letech. Pro rozdělení anotací kategorií pohlaví a emocí byly použity první písmena jejich anglických překladů. Pro emoce byly zvoleny 4 klasifikační třídy, konkretně se jedná o S (smutek), H (štěstí), N (neutrálnost) a A (rozzlobenost). Pohlaví bylo anotováno písmeny M (muž) a F (žena). V databázi je rovněž uložen list souřadnic, kde x_1, y_1 symbolizují souřadnice levého horního rohu a x_2, y_2 symbolizují souřadnice pravého spodního rohu snímku.

Z obrázku 4.1 lze vyčíst, že v kategorii věk osobností se podařilo úspěšně dosáhnout rovnoměrného rozložení dat. Jelikož je převážná většina slavných osobností v databázi z filmového průmyslu, tak data s informací o věku korespondují s pravděpodobnostním rozložením věku herců/hereček. V kategorii emoce výrazným způsobem převažuje neutrálnost, což negativně ovlivnilo výsledky v úloze rozpoznání emocí.



Obrázek 4.1: Rozložení anotací vytvořené databáze

4.2 Použitý hardware

V případech, kdy jsou uvedeny statistiky časové náročnosti, byl využit notebook s následující hardwarovou specifikací:

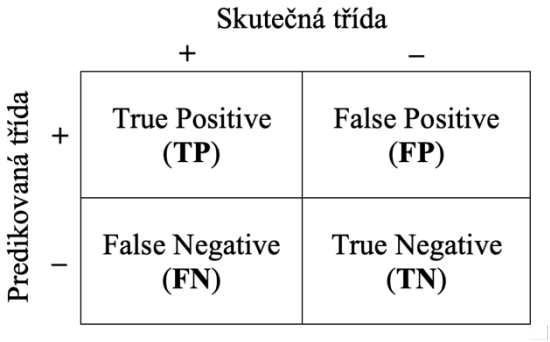
- CPU: AMD Ryzen 5 4500U, 2,3GHz
- GPU: integrovaná AMD Radeon Graphics
- RAM: 16GB

Výkon stroje však nebyl dostačující, aby bylo možné trénovat hlubší neuronové sítě (YOLO), jelikož nebyla k dispozici grafická karta podporující technologii CUDA. Pro testování těchto metod bylo využito prémiové verze Google colab, který umožňuje hostované spouštění Jupyter notebooků. Přiřazený hardware zde není vždy stejný, ale pohybuje se na podobné úrovni:

- GPU: NVIDIA A100-SXM, 40 GB
- RAM: 52-90GB

4.3 Metriky vyhodnocení

Vyhodnocení metod není vhodné provádět pouze na základě úspěšnosti, protože dostatečně nepopisuje silné či naopak slabé stránky detektoru a klasifikátoru. V případě, kdy je třeba zjistit citlivost nebo přesnost výsledků klasifikace, je využito metrik precison, recall a F1 skóre. Pro jejich výpočet je nutné sestavení tzv. matice záměn



Obrázek 4.2: Matice záměn binární klasifikace

kde TP značí, že predikovaná kladná třída odpovídá skutečné. FP naopak značí zápornou, ale neodpovídající predikci. Pokud je predikována záporná hodnota odpovídající skutečné, je tato situace označena jako TN. FN značí provedení negativní predikce, přestože je skutečná třída pozitivní. Z matice záměn je možné spočítat odpovídající metriky. Precision udává míru přesnosti všech provedených predikcí, tedy jak moc z provedených predikcí je opravdu správných.

$$P = \frac{TP}{TP + FP} \tag{4.1}$$

Recall odpovídá poměru správných predikcí vůči všem pozitivním.

$$R = \frac{TP}{TP + FN} \tag{4.2}$$

Metrika F1 skóre je kombinací precision a recall označující kvalitu klasifikace.

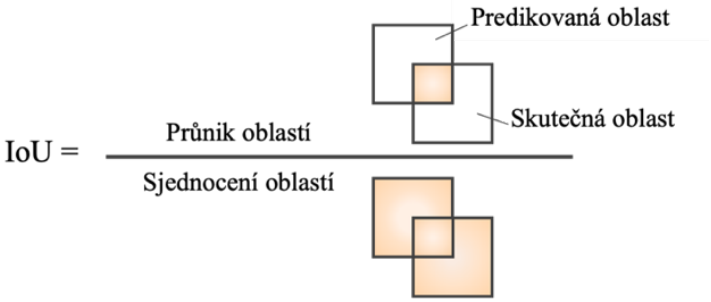
$$F1 = \frac{2 \times P \times R}{P + R} \tag{4.3}$$

4.3.1 IoU

IoU (Intersection over Union), též nazývané Jaccardův index, udává míru překrytí mezi skutečnou oblastí A a detekovanou oblastí B. Je dáno vztahem

$$J(A, B) = IoU = \frac{A \cap B}{A \cup B} \tag{4.4}$$

kde $IoU \in [0, 1]$ a prahovou hodnotou pro označení detekce jako skutečně pozitivní (tedy TP) nejčastěji bývá $0,5$.



Obrázek 4.3: Znázornění vztahu pro výpočet Jaccardova indexu

4.3.2 Mean average precision

Metrikou používanou pro vyhodnocení kvality detektorů je mAP (mean average precision). K jejímu určení je potřeba vypočíst hodnotu AP (Average Precision), která je definována jako plocha pod Precision-Recall (dále PR) křivkou.

$$AP = \int_{r=0}^{1} p(r)dr \tag{4.5}$$

Aby bylo možné PR křivku vykreslit, je nutné provést testování detektoru (vytvoření matice záměn) pro různé prahové hodnoty (např. IoU). Ke každé z x prahových hodnot je vypočten precision a recall, čímž je vztah zjednodušen na

$$AP = \frac{1}{x} \sum_{r=0}^{1} p(r) \tag{4.6}$$

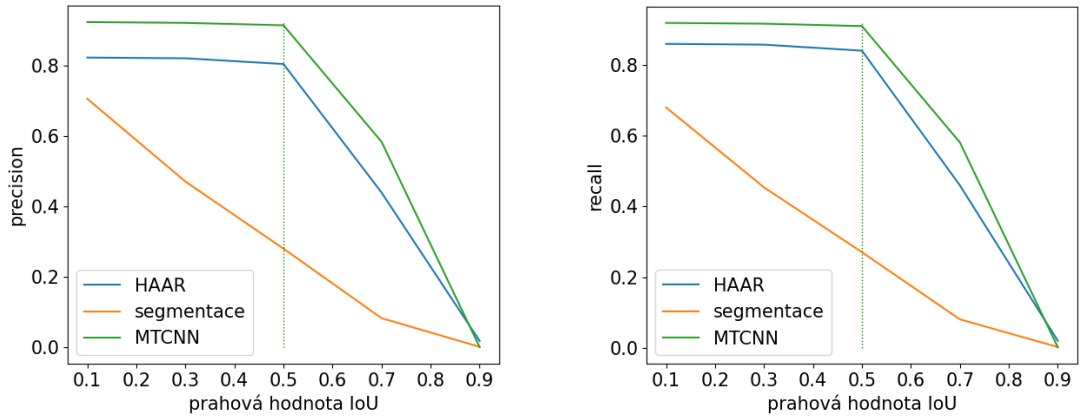
Pokud je vyhodnocována pouze jedna třída, je AP zároveň mAP. V případě N tříd dochází ke zprůměrování AP všech tříd [15]

$$mAP = \frac{1}{N} \sum_{i=0}^{N} AP_i \tag{4.7}$$

4.4 Detekce obličeje

Pro účely porovnání detektorů bylo nutné stanovit minimální hodnotu IoU, při které lze akceptovat výsledek detekce jako správný. Z testování optimální prahové hodnoty byla vynechána metoda HOG, jelikož byla využita jako výchozí detektor při anotaci databáze, tudíž by nereflektovala změnu prahu IoU.

Na výsledcích testování je pozorovatelné, že se snižováním prahové hodnoty IoU stoupá precision i recall detektorů, příčinou je absence negativních snímků. Zvyšováním nároku na překrytí plochy mezi predikovanou a skutečnou oblastí je pouze snižován počet TP, důsledkem čehož stoupá počet FN, FP. Volba prahové hodnoty IoU byla stanovena na $0,5$, jelikož vykazuje nejlepší poměr mezi úspěšností a skutečnou přesností.



(a) na precision (b) na recall

Obrázek 4.4: Vliv změny prahové hodnoty IoU

Tabulka 4.1: Tabulka porovnání výsledků metod vzhledem ke změnám prah. hodnoty

Práh IoU	Precison			Recall		
	MTCNN	HAAR	Segment.	MTCNN	HAAR	Segment.
0,1	0,924	0,823	0,706	0,92	0,8606	0,680
0,3	0,922	0,821	0,471	0,917	0,8584	0,454
0,5	0,915	0,805	0,281	0,911	0,8414	0,270
0,7	0,584	0,440	0,083	0,581	0,46	0,080
0,9	0,001	0,018	0,002	0,001	0,0196	0,002

Nejlepších výsledků dosahuje metoda MTCNN s hodnotou F1 skóre $0,915$. Z tabulky 4.2 lze vyčíst, že pro metodu MTCNN je precision a recall téměř identický. Metoda Viola-Jones (v práci označována jako HAAR) dosahuje velmi solidních výsledků s celkovou hodnotou F1 skóre $0,823$, přesto je stále viditelný vidět mírný pokles, především v metrice precision. Nižší hodnota precision je způsobena vysokým počtem FP, protože detektor viola-jones je náchylnější k chybné pozitivní predikci okolních objektů. Avšak tuto chybovost kompenzuje velmi vysokou rychlostí $5,325$ snímků za vteřinu, což je zhruba $6\times$ rychlejší než metoda MTCNN. Metoda segmentace prahováním dosahuje přijatelných výsledků pouze dokud $IoU \geq 0,3$, poté je detekční kvalita velmi nízká, přestože rychlost dosahuje nejvyšší hodnoty ze všech analyzovaných metod.

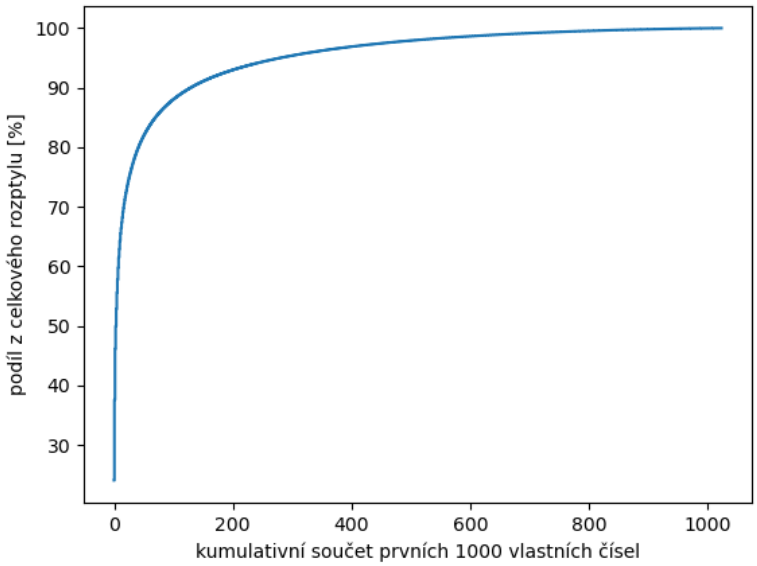
Tabulka 4.2: Srovnání metod při prahové hodnotě IoU= 0,5

Název metody	Precision	Recall	Úspěšnost	F1	FPS [s]
MTCNN (CPU)	0,915	0,911	0,840	0,913	0,877
HAAR	0,805	0,8414	0,699	0,823	5,325
Segmentace	0,281	0,270	0,159	0,275	26,31

4.5 Rozpoznání osoby

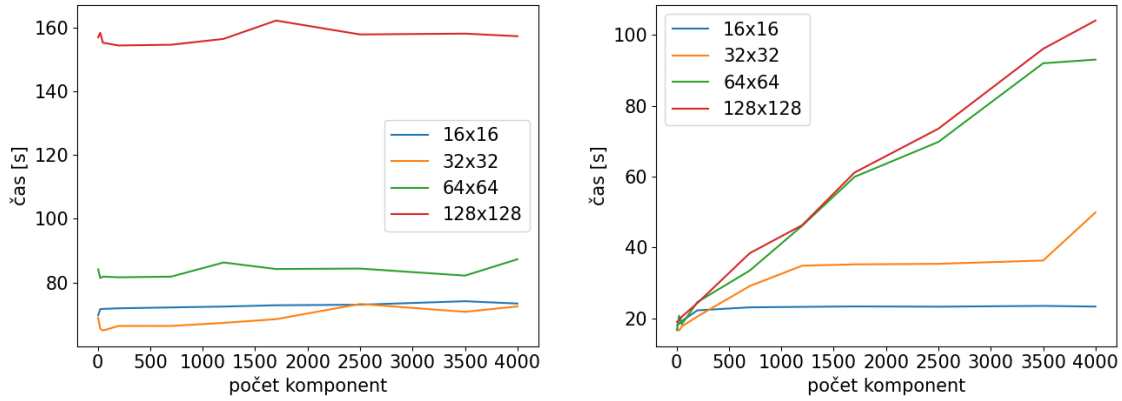
4.5.1 PCA

Metoda PCA byla prvně testována vlastní implementací, avšak z důvodu pomalé rychlosti byla nakonec zvolena implementace knihovny scikit-learn. První analyzovanou vlastností byla závislost počtu zvolených hlavních komponent na úspěšnost a rychlost trénování/testování.



Obrázek 4.5: Podíl vlastních čísel na celkový rozptyl dat

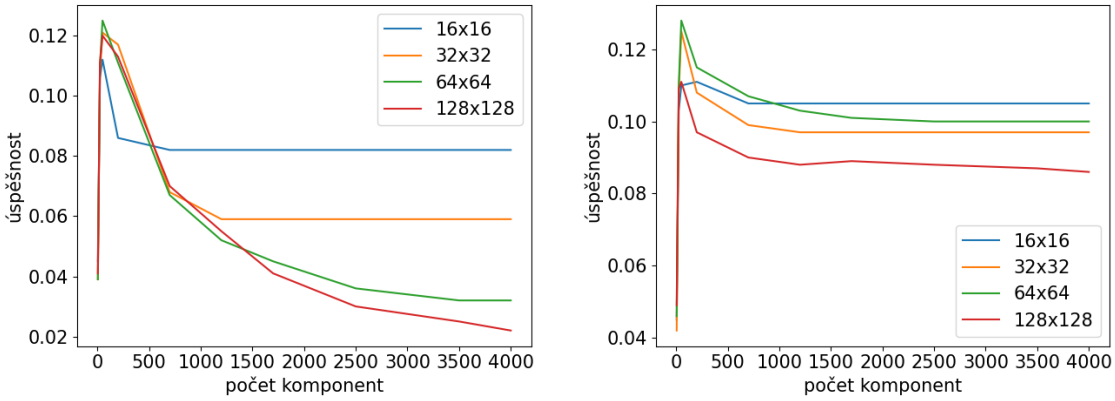
Testování probíhalo pro různé velikosti snímků v rozsahu 16×16px až 128×128px.



(a) trénování (b) testování

Obrázek 4.6: Závislost velikosti obrázku a počtu komponent na rychlost

Z výsledků lze konstatovat, že doba trénování není závislá na počtu komponent (výpočet vlastních čísel je před volbou počtu komponent). S velikostí obrázků lineárně roste doba testování, jelikož roste velikost kovarianční matice a tím doba výpočtu vlastních čísel a vektorů (složitost $O(n^3)$). Doba testování roste jak s velikostí snímků, tak s počtem zvolených hlavních komponent. Je vhodné zvolit takový počet komponent, který vykazuje vysokou úspěšnost a zároveň co nejvíce redukuje dimenzi dat.



(a) manhattanská vzdálenost

(b) euklidovská vzdálenost

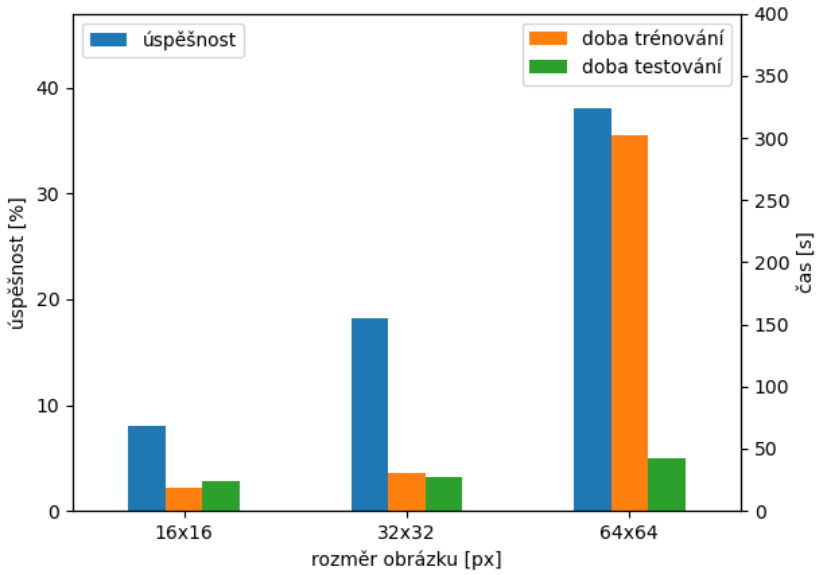
Obrázek 4.7: Vliv počtu zvolených komponent a velikosti snímku na úspěšnost

Při vyhodnocení úspěšnosti bylo testováno využití manhattanské a euklidovské vzdálenostní metriky. Euklidovská vzdálenost vykazuje stabilnější rozpoznávací schopnost i pro větší velikosti snímku a vyšší počet komponent. Nejvyšší úspěšnosti bylo dosaženo při volbě menšího počtu hlavních komponent. Při volbě více než 50 komponent úspěšnost rozpoznávání nepatrně klesá. Přesto úspěšnost PCA přesahuje pouze nepatrně 12%.

4.5.2 MACE

Algoritmus Mace byl obdobně otestován pro různé velikosti vstupních snímků, v tomto případě však bylo vynecháno testování snímků o velikosti 128 × 128px. Výpočet inverzních matic a následného korelačního Mace filtru v trénovací části byl výpočetně příliš náročný a nepodařilo se tento test z časové náročnosti úspěšně provést.

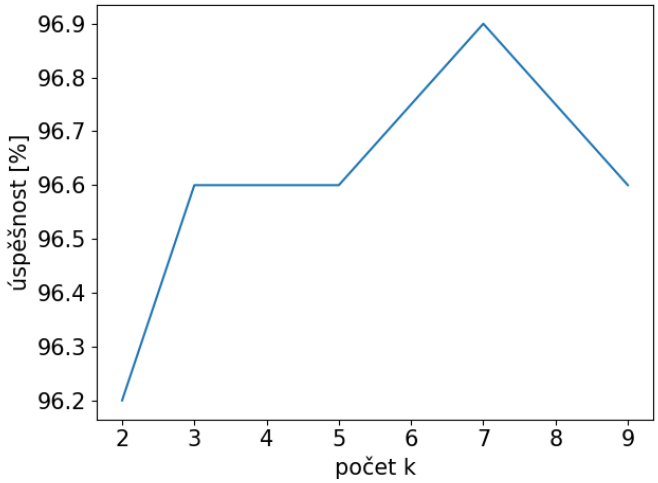
Úspěšnost algoritmu Mace byla nejvyšší pro velikost obrázku 64 × 64px, kde dosahoval úspěšnosti 38,1%. Pro menší velikosti snímku velmi rychle klesá i úspěšnost rozpoznávání. Na obrázku 4.8 je patrný prudký nárůst doby trénování při velikosti snímku 64 × 64px oproti 32 × 32px. Doba testování se mění pouze zanedbatelně v závislosti na velikosti snímku.



Obrázek 4.8: Srovnání výsledků při různé velikosti snímků

4.5.3 DML

Metoda DML nevyžaduje normalizaci velikosti snímků, proto byl tento test vynechán. Při testování byla zvolena klasifikace metodou nejbližšího souseda (K-NN) a porovnána úspěšnost při volbě různých k-nejbližších sousedů euklidovskou vzdáleností.



Obrázek 4.9: Závislost volby k-nejbližších sousedů na úspěšnost

Lze vyčíst, že volba k nehraje téměř žádnou roli na výslednou úspěšnost, která se mění pouze nepatrně. Nejvyšší úspěšnosti $96,9\%$ bylo dosaženo při $k = 7$, avšak

výhodnější se jeví využití menšího počtu k z důvodu nižší časové náročnosti.

4.5.4 Shrnutí

Nejúspěšnější metodou byla neuronová síť s architekturou DML, která vykazovala úspěšnost $96,9\%$. Přes takto vysokou úspěšnost, je tato metoda limitována velmi vysokou časovou náročností, s kterou bylo třeba počítat při návrhu aplikace pro testování v reálném čase. Při testování DML byla zvolena klasifikace metodou nejbližšího souseda (KNN), kde nejúspěšnější se ukázalo využití $k = 7$ a pro porovnání vzdálenosti euklidovská metrika. Nejvyšší úspěšnosti $38,1\%$ dosahovala metoda MACE při velikosti snímků 64×64. Použití PCA se ukázalo i přes rychlé trénování a testování jako velmi neúspěšné. Nejvyšší úspěšnosti bylo dosaženo při využití prvních 50 komponent, euklidovské vzdálenosti a velikosti snímku 64×64.

Tabulka 4.3: Srovnání metod identifikace osob

Metoda	TP	FP	Doba trénování [s]	Doba testování [s]	Úspěšnost [%]
DML	962	38	1218,99	322,27	96,9%
MACE	381	619	302,56	42,83	38,10%
PCA	128	872	81,72	20,98	12,80%

4.6 YOLO v7

K analýze metody YOLOv7 bylo využito implementace autora WongKinYiu, která umožňuje snadné trénování/testování skrze příkazovou řádku.

Obrázek 4.10: Příklad aplikace YOLO detektoru pro rozpoznání pohlaví

4.6.1 Příprava dat

Vstupním snímkům byla potřeba změnit velikost na 640×640 a převést formát ohraničujících boxů vytvořené databáze na formát podporovaný YOLO sítí. Tento

formát je definován ve tvaru $x_{centrum}, y_{centrum}$, šířka w a výška h. Odlišností oproti běžným formátům je normalizace těchto souřadnic na relativní v rámci buňky v intervalu $[0, 1]$.

Následně bylo možné úpravou konfiguračních souborů změnit hyperparametry sítě, ponechané na výchozích hodnotách. Při spuštění skriptu byly nastaveny pouze:

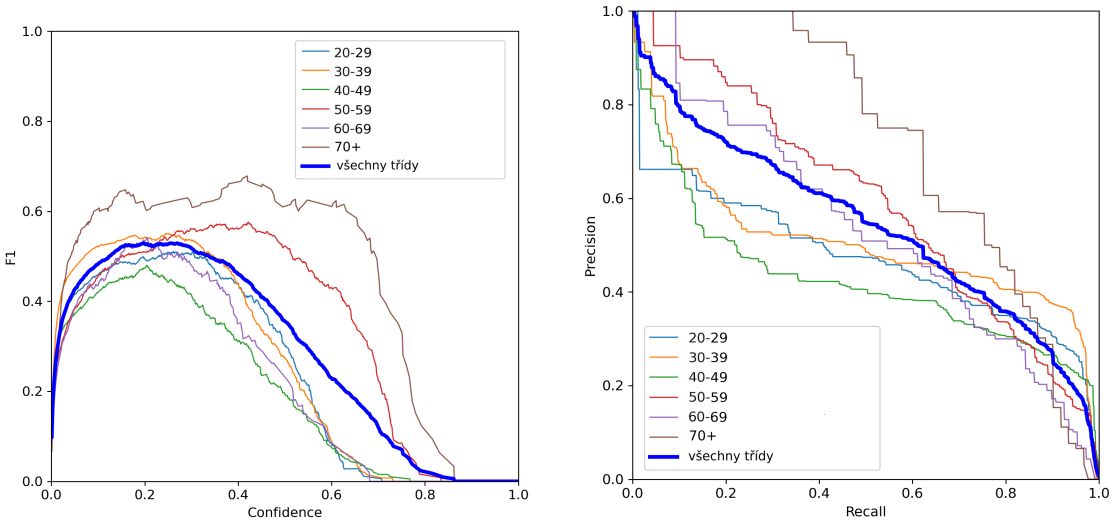
- Batch size : 64
- Počet epoch: 20 (pohlaví), 50 (emoce), 70 (věk)

Finální počet epoch u jednotlivých úloh byl volen různě, jelikož z analýzy předchozích testování byla vypozorována epocha od které nadále nedochází k výraznému zlepšení výsledků validační sady a mohlo by dojít k přetrénování.

4.6.2 Rozpoznávání věku

Rozpoznání věku má ze všech testovaných úloh nejvyšší počet klasifikačních tříd. Zároveň rozdíly mezi třídami jsou často velmi minimální, proto je nejvyšší hodnota F1 skóre pokud $c \in [0.15, 0.4]$. Pro vyšší hodnoty confidence již kvalita modelu velmi strmě klesá.

Vlivem náhodného rozdělení databáze nastala situace, že třída 10-19 let nebyla zastoupena ve validační sadě. Autoři původní implementace s touto situací nejspíše nepočítali, protože vznikla chybná indexace tříd znázorněných v legendě vygenerovaných grafů. Namísto testovaných tříd obsahovala pouze prvních 6 tříd zapsaných v konfiguračním souboru. Legenda musela být manuálně upravena na správné popisky tříd a autoři byli upozorněni na chybu v implementaci algoritmu.



(a) F1 křivka (b) PR křivka

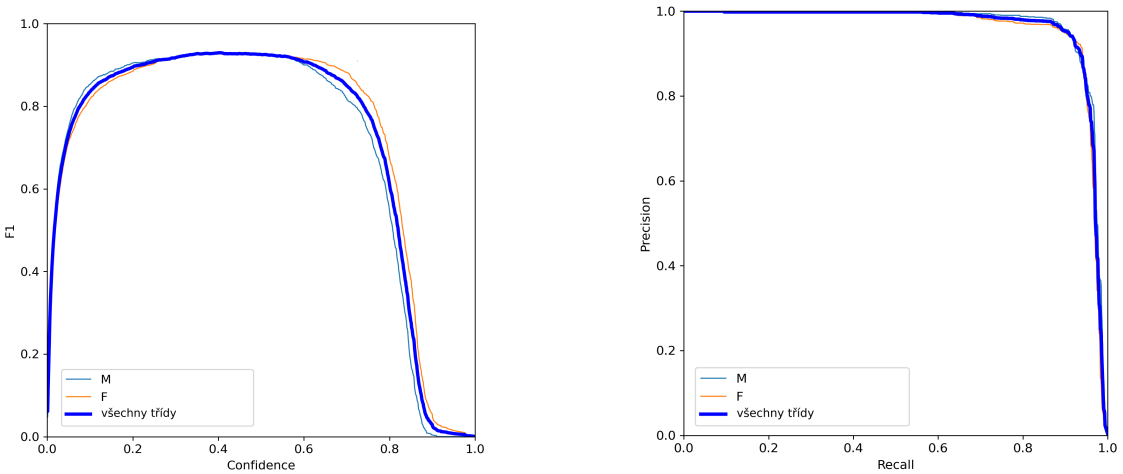
Obrázek 4.11: Křivky natrénovaného modelu rozpoznávání věku

Tabulka 4.4: Vyhodnocení metriky mAP0.5

Třída	AP@0.5
20-29	0,473
30-39	0,506
40-49	0,423
50-59	0,591
60-69	0,539
70+	0,722
Všechny třídy	0,543 mAP@0.5

4.6.3 Rozpoznávání pohlaví

Jedná se o binární kategorii s rovnoměrně rozloženými daty, kde třídy jsou od sebe příznakově (i vizuálně) velmi odlišné. Tím jsou splněny předpoklady pro velmi úspěšné rozpoznávání, čemuž odpovídají výsledky. Křivky modelu dosahují téměř ideálních tvarů, jak poukazuje celkové mAP@0.5 s hodnotou 0,961.



(a) F1 křivka

(b) PR křivka

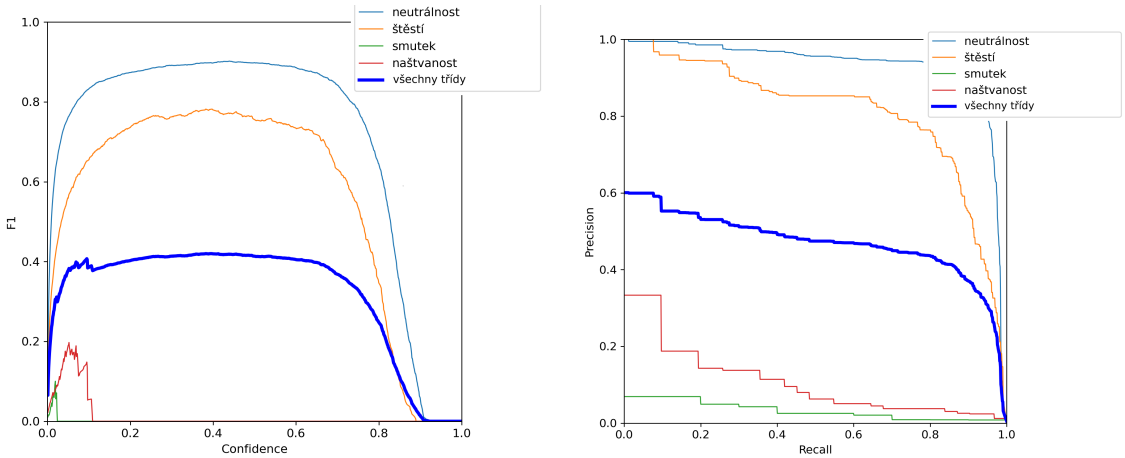
Obrázek 4.12: Křivky natrénovaného modelu rozpoznávání pohlaví

Tabulka 4.5: Vyhodnocení metriky mAP0.5

Třída	AP@0.5
Muž	0,966
Žena	0,955
Všechny třídy	0,961 mAP@0.5

4.6.4 Rozpoznávání emocí

Z důvodů uvedených v kapitole 4.1 bylo trénování detektoru pro tuto úlohu velmi náročným úkolem. Z grafů na obrázku 4.13 je možné vyčíst, že neutrálnost a štěstí dosahují vysokých hodnot F1 skóre i při vyšší hodnotě confidence. To neplatí pro zbylé dvě třídy, které při takto malém zastoupení v trénovací (i testovací) sadě nebylo možné správně natrénovat a celkový výsledek detektoru je jimi ovlivněn. Již od nízké confidence je precision velmi vysoký (dokonce roven 1) a recall velmi rychle klesá k nule. To ukazuje sice velmi vysokou důvěru v TP snímky, avšak velmi nízkou rozpoznávací schopnost detektoru s poměrově vysokým zastoupením FN oproti TP.



(a) F1 křivka

(b) PR křivka

Obrázek 4.13: Křivky natrénovaného modelu rozpoznávání emocí

Celkové mAP@0.5 pro všechny třídy je tedy 0,474, i přes to, že neutrálnost a štěstí dosahují této hodnoty větší než 0,8. Závěrem lze říci, že detektor je možné použít pro poměrně spolehlivé rozpoznávání, zda-li je osoba na obrázku s neutrálním výrazem či šťastná.

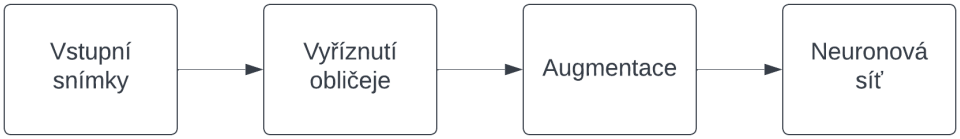
Tabulka 4.6: Vyhodnocení metriky mAP0.5

Třída	AP@0.5
Neutrálnost	0,933
Štěstí	0,811
Naštvanost	0,106
Smutek	0,032
Všechny třídy	0,474 mAP@0.5

4.7 Klasifikace využitím neuronových sítí

Pro úlohy rozpoznání věku a pohlaví byl vytvořen jupyter notebook *own_cnn.ipynb* umožňující otestovat libovolnou neuronovou síť navrženou v knihovně PyTorch. Da-

tabáze osob byla rozdělena na 3000 trénovacích, 1000 validačních a 1000 testovacích snímků.



Obrázek 4.14: Proces předzpracování dat

Ze vstupních snímků byla nejprve oříznuta obrazová oblast obličeje daných osob a následně proběhlo předzpracování následujícími operacemi:

- změna velikosti na 256×256px
- vyříznutí vnitřních 224×224px
- normalizace odečtením průměrné střední hodnoty a rozptylu

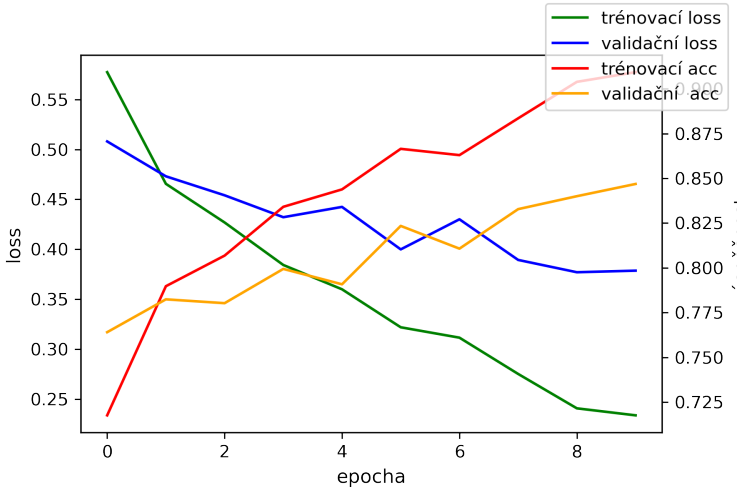
Hodnoty hyperparametrů byly laděny na validační sadě a nakonec shodně použity pro všechny architektury sítí:

- počet epoch = 10
- batch size = 64
- learning rate = 0,001
- momentum = 0,9
- optimalizační metoda = SGD
- kriteriální funkce = cross-entropy loss

Původně byla zvolena optimalizační metoda Adam (Adaptive momentum), která vykazuje rychlejší konvergenci k minimu funkce. Po naladění vhodných parametrů však vykazovala vyšší úspěšnost metoda SGD, proto byla i přes pomalejší konvergenci zvolena.

4.7.1 Průběh trénování

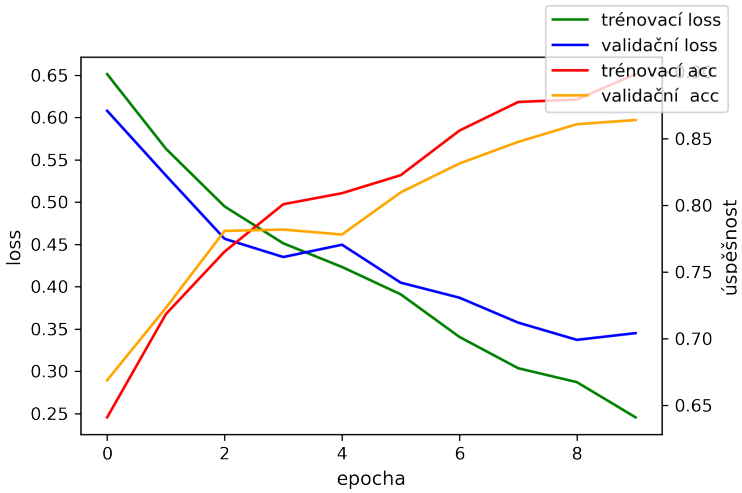
Celkem byly zvoleny 4 architektury neuronových sítí, které byly nejprve natrénovány pro úlohu rozpoznání pohlaví. Nejjednodušší zvolenou architekturou byla síť typu vícevrstvý perceptron (MLP) s jednou skrytou lineární vrstvou následovanou dropout vrstvou s pravděpodobností vynulování uzlu rovnou $0,4$ (omezuje vliv přetrénování) a batch normalizací. Aktivační funkce pro všechny architektury byla zvolena ReLU.



Obrázek 4.15: Průběh trénování MLP

Přestože se je jedná pouze o jednoduchou neuronovou síť, lze z průběhu vyčíst poměrně vysoká úspěšnost. Je ale nutné konstatovat, že se jedná o binární klasifikaci, kde jednotlivé třídy jsou rovnoměrně zastoupené. Náhodný binární klasifikátor by tedy dosahoval úspěšnosti zhruba 50%.

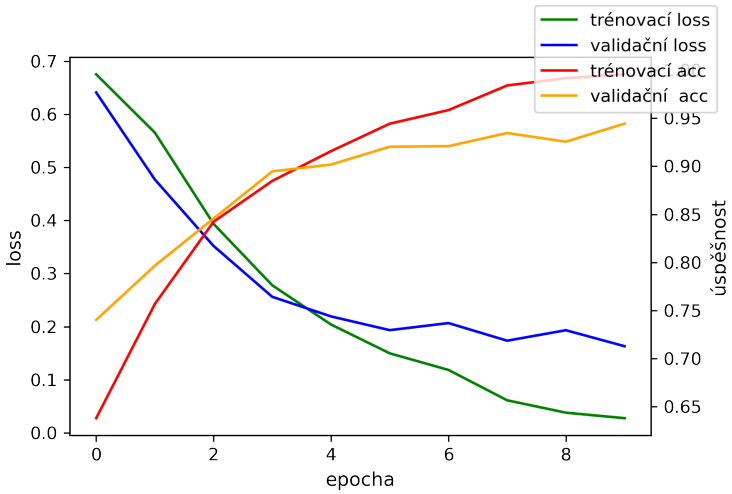
Jelikož se jedná o úlohy zpracování obrazu, lze předpokládat dosáhnutí nejvyšší úspěšnosti s použitím konvolučních sítí, proto jsou zbylé neuronové sítě zaměřeny právě na ně. Druhou architekturou byla CNN s dvěmi konvolučními vrstvami, maxpooling s 2×2 filtrem a na výstup připojenými 2 plně propojenými vrstvami.



Obrázek 4.16: Průběh trénování dvouvrstvé CNN

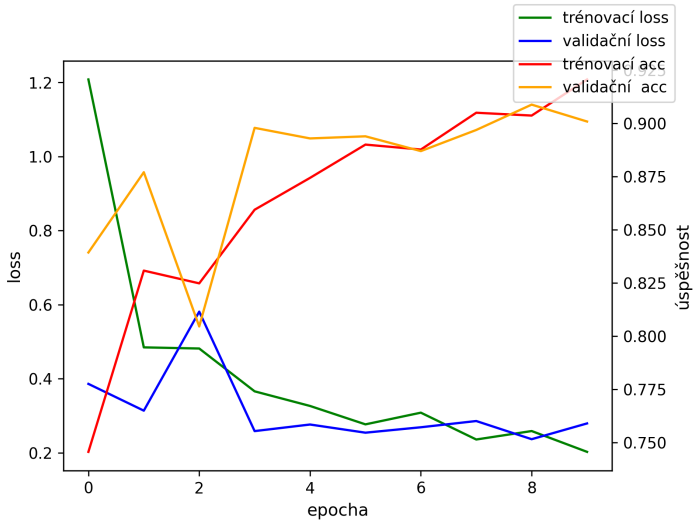
Dvouvrstvá CNN dosahuje podobného výsledku jako MLP, přestože je možné sledovat mírně pomalejší konvergenci. To by mohlo být vylepšeno úpravou hyperparametrů learning rate a momentum. V rámci této práce byly však hyperparametry zachovány shodné, jelikož následující sítě by bylo časově velmi náročné ladit bez možnosti využití grafické karty podporující technologii CUDA, tím pádem by byly

znevýhodněny jejich výsledky. Proto byla volba hyperparametrů provedena na základě nejúspěšnějších výsledků výše zmíněných sítí. Zbylé dvě sítě vycházejí z již známých architektur VGG7BN a ResNet9. Architektura VGG7BN byla rozšířena o jeden blok (conv-BN-ReLU-MP, zkráceně CNRM) a přidána jedna FC vrstva na výstupu. ResNet9 byl naopak zjednodušen odstraněním jednoho reziduální bloku ($z = x + res(CNRM, CNRM)$), z důvodu zvýšení rychlosti trénování.



Obrázek 4.17: Průběh trénování VGG7BN

Z výsledku je patrné, že použití komplexnější architektury VGG7BN přineslo výrazné zvýšení úspěšnosti na validační sadě. Úspěšnost trénovací sady se již u 10 epochy limitně blíží 100% a v průběhu následujících epoch by mohlo dojít k přetrénování.



Obrázek 4.18: Průběh trénování ResNet9

Poslední byla natrénována modifikovaná verze sítě ResNet9, která je i přes mírné odlehčení stále komplexnější než předcházející VGG7BN. Průběh trénování indikuje

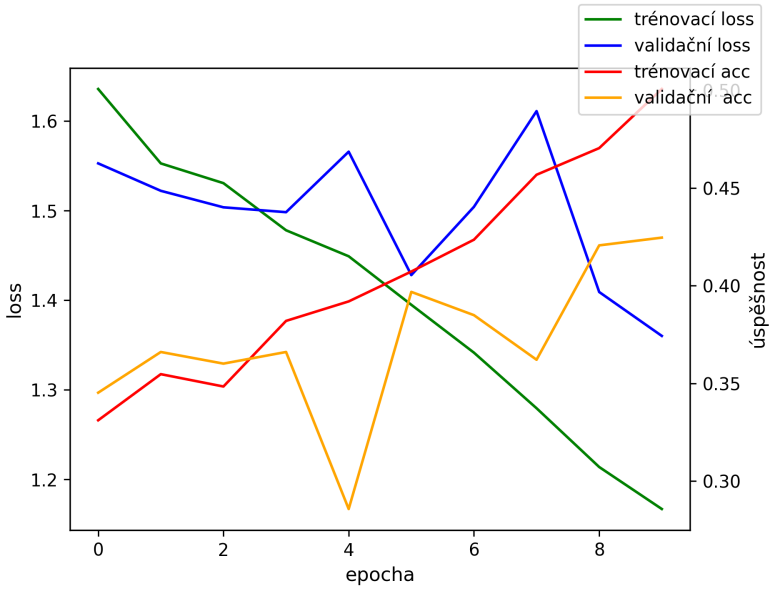
nalezení lokálního minima a výsledná úspěšnost na validační sadě je mírně horší, než tomu bylo u předcházející sítě.

Vyhodnocení natrénovaných sítí proběhlo na testovací sadě databáze. Nejúspěšnější architekturou se ukázala VGG7BN s celkovou úspěšností 89, 58%. Podobných výsledků v úspěšnosti i časové náročnosti dosahovali MLP a dvouvrstvá CNN. ResNet i přes svou komplexitu zaostával oproti VGG7BN o necelé dvě procenta avšak s mnohem vyšší časovou náročností testování.

Tabulka 4.7: Srovnání neuronových sítí na testovací sadě

Název sítě	loss	úspěšnost [%]	doba trvání [s]
MLP	0,4198	81,66	30,27
dvouvrstvá CNN	0,379	82,96	32,78
VGG7BN	0,291	89,58	90,21
ResNet	0,343	87,83	260,90

Trénování rozpoznávání věku se ukázalo velmi obtížnou úlohou. Sítě MLP a dvouvrstvá CNN se chovají téměř jako náhodný klasifikátor (zohledňující zastoupení tříd). Jejich úspěšnost je pro 10 epoch téměř konstantní. Síť Resnet se podařilo natrénovat s úspěšností na validační sadě dosahující 40%, při analýze na testovací sadě však tato úspěšnost klesla na pouhých 29, 7%. Síť VGG7BN vykazovala i pro tuto úlohu lepších výsledků, kdy úspěšnost na validační sadě dosáhla 42% a 34, 35% na sadě testovací. Pokus byl pro VGG7BN opakován, tentokrát s 20 epochami. Tento test byl neúspěšný, jelikož i přes 53% úspěšnosti na validační sadě došlo k přetrénování modelu, který na testovacích datech dosáhl úspěšnosti pouhých 33%.



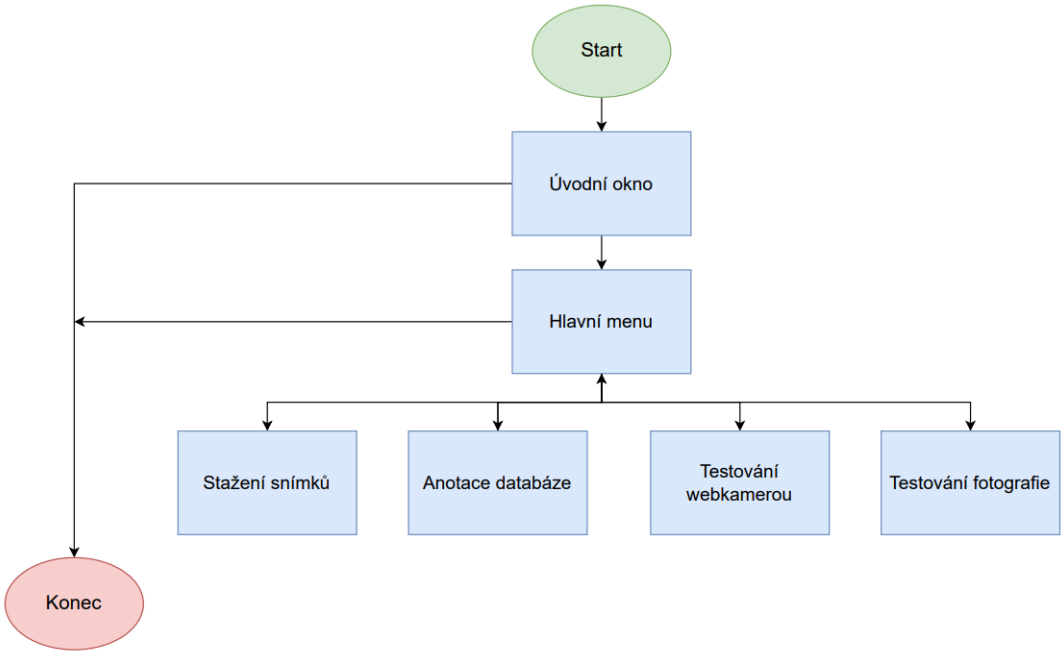
Obrázek 4.19: Průběh trénování rozpoznávání věku VGG7BN

5 Aplikace

Pro usnadnění sestavení databáze a následné vizuální testování byla vytvořena multiplatformní aplikace podporující Windows, Linux a MacOS. K jejímu návrhu byl využit framework Kivy, napsán v programovacím jazyce Python. Grafické uživatelské rozhraní (GUI) je tvořeno v jazyce kvlang (kivy language) připomínající strukturou a použitím kaskádové styly. Tyto technologie byly zvoleny kvůli snadné propojitelnosti s Python kódem, ve kterém jsou implementovány knihovny využité pro testování výše zmíněných metod.

5.1 Struktura aplikace

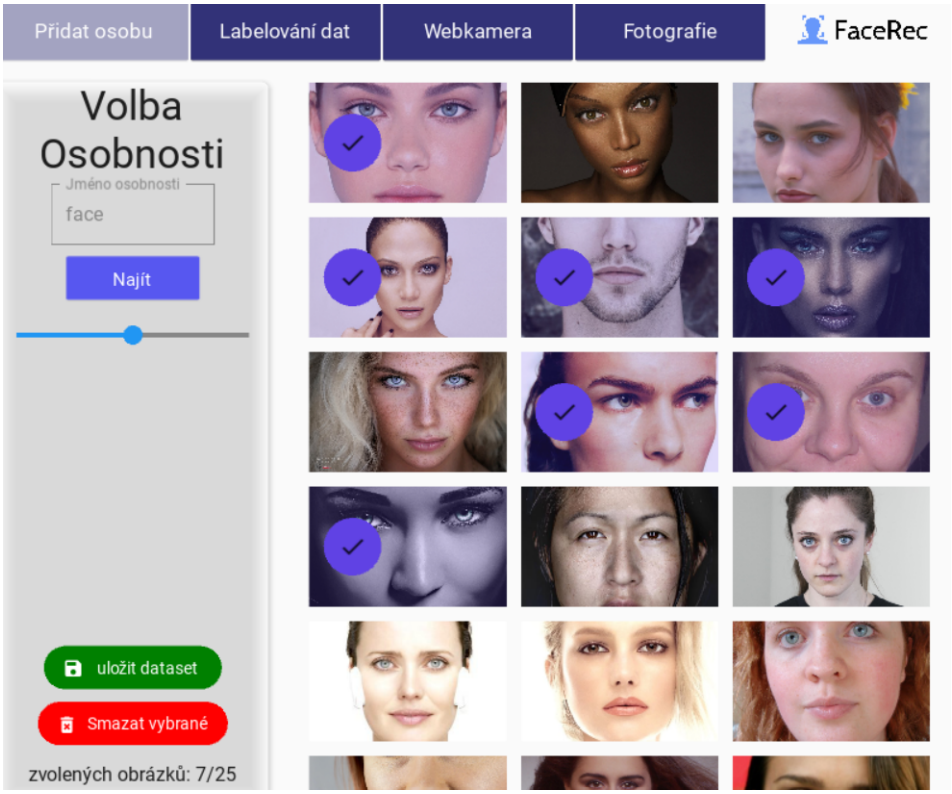
Aplikace se děli na 4 hlavní okna



Obrázek 5.1: Vývojový diagram aplikace

5.1.1 Stažení snímků do databáze

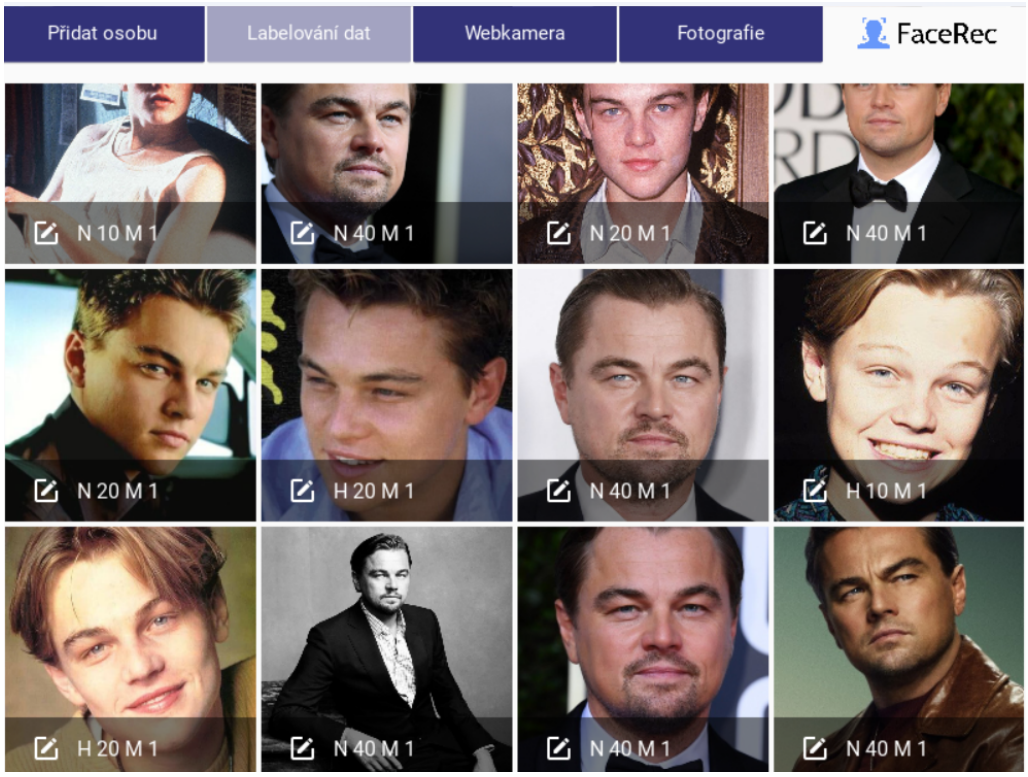
Toto okno slouží pro stažení snímků a následné uložení snímku do databáze. Napsáním klíče (jména) do textového pole a stisknutím tlačítka *najít* je staženo n snímku do adresáře *Osobnost/jméno*. Počet stažených snímku je volen posuvníkem, který je limitován intervalem $[0, 50]$. Tento limit byl volen z důvodu časově náročného (až desítky vteřin) stahování při větším počtu zvolených snímků. Následně je nabídnuta možnost zvolení snímků, které si uživatel nepřeje uložit do své databáze. Jejich označením a stiskem tlačítka *Smazat vybrané* dojde k jejich odstraněné. Tlačítkem *Uložit dataset* jsou veškeré nadále zobrazené fotografie uloženy do databáze ve formátu JSON a inicializovány prázdné anotace. Jedinou inicializovanou anotací je jméno, které je zvoleno klíčem napsaným v textovém poli.



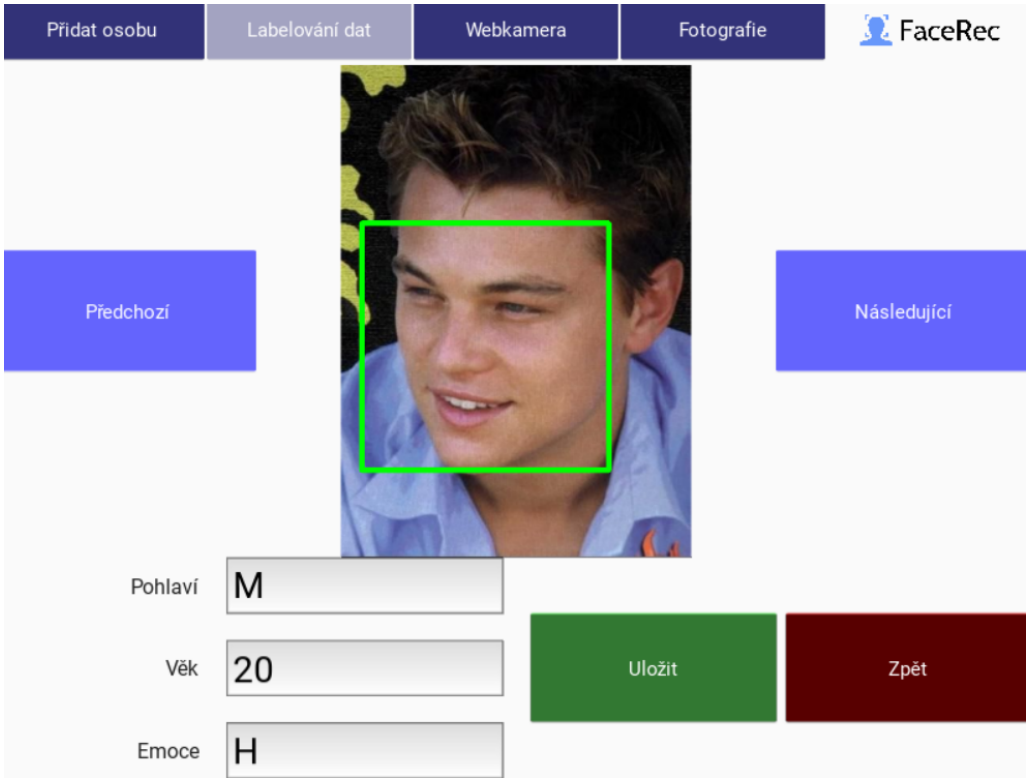
Obrázek 5.2: Stažení snímku do databáze

5.1.2 Anotace dat

Okno s funkcí anotace databáze je rozděleno na tři samostatná podokna. V prvním podokně je možnost volby osobnosti, kterou si uživatel přeje upravit. Následně jsou zobrazeny všechny fotografie dané osoby včetně anotací, pokud jsou již přiřazeny. Funkce anotace obličeje není implementována v aplikaci, ale byl vytvořen samostatný skript */label_facearea.py* sloužící k tomuto účelu. V posledním podokně je již možné upravovat anotaci jednotlivých snímků patřících dané osobě.



(a) zobrazení snímků zvolené osoby

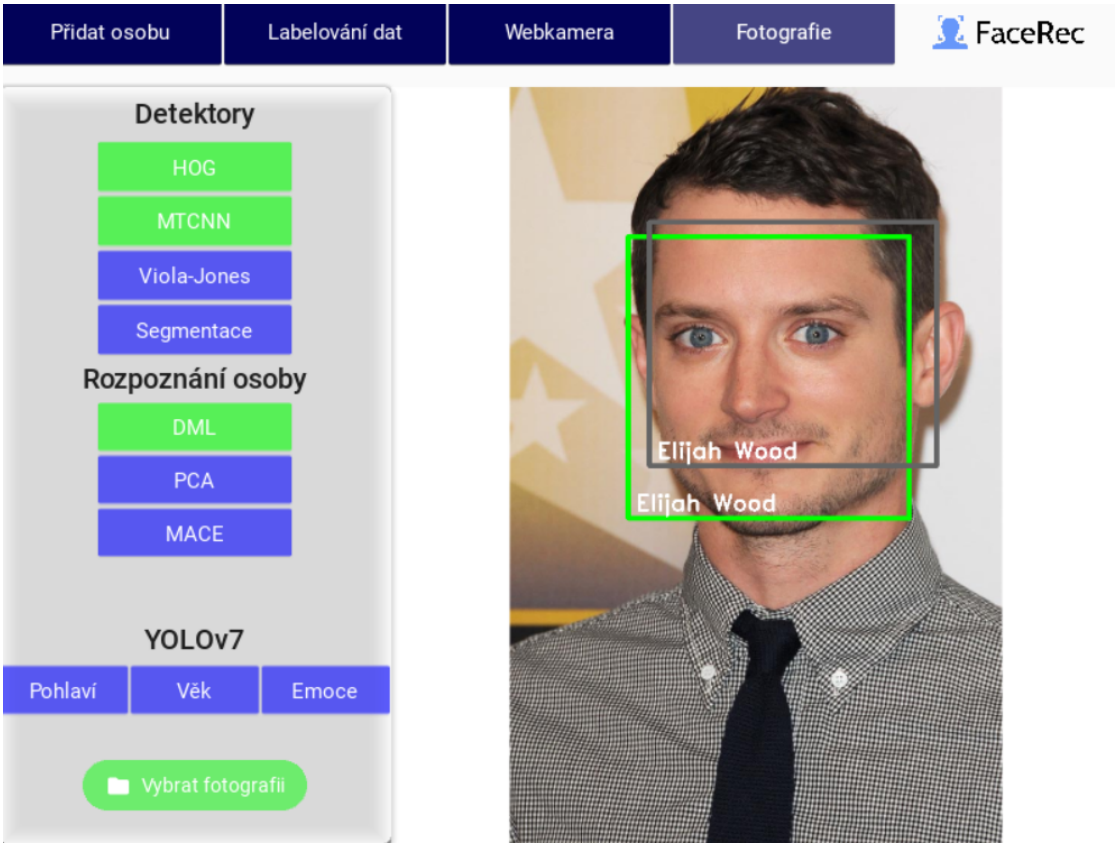


(b) úprava anotací

Obrázek 5.3: Okna anotace databáze

5.1.3 Vizuální testování na libovolné fotografii

Další funkcí aplikace je vizuální testování metod zmíněných v teoretické části práce. Stiskem tlačítka *Vybrat fotografii* je zobrazena možnost vybrat libovolnou fotografii nacházející se v paměti zařízení. Při testování je nutné dodržet několik pravidel, v opačném případě aplikace nespustí požadovanou metodu. Pro identifikaci osob platí, že není možná aktivace více metod současně a je vyžadováno využité jedné či více detekčních metod. Dále je možné otestovat rozpoznávání pohlaví, věku či pohlaví natrénovanou sítí Yolov7. Aktivací jedné z nabízených možností dojde asynchronně ke spuštění skriptu *detect.py*. Výstupem tohoto skriptu je stažení rozpoznaného snímku do adresáře *runs/detect/exp* a následnému zobrazení v aplikaci. Minimální prahová hodnota confidence zobrazení byla nastavena na hodnotu $0,2$. Takto nízká hodnota byla zvolena, jelikož při vyšších již nedocházelo k rozpoznání, především v úloze rozpoznání věku.



Obrázek 5.4: Příklad vizuálního testování snímku

5.1.4 Vizuální testování webkamerou

Pokud má uživatel k dispozici připojenou videokameru je možné otestovat vybrané metody v reálném čase. Obraz je aktualizován s obnovovací frekvencí 30 snímků za sekundu. Identifikační algoritmy avšak není možné aplikovat při každém obnovení okna, proto je provedena aktualizace identifikace každé 3 vteřiny.

Závěr

V rámci této bakalářské práce byl vytvořen systém pro automatické detekovaní obličeje, identifikaci osob a rozpoznávání emocí, věku a pohlaví v digitálním obraze. Algoritmy těchto úloh je možné otestovat v reálném času skrze připojenou webkameru nebo na libovolné fotografii. Systém dále disponuje nástroji pro stažení a následnou anotaci v kategorii emocí, věku a pohlaví. Pro účely testování byla sestavena databáze tvořena 100 unikátními osobnostmi po 50 různých snímcích. Ta se nicméně neukázala vhodná pro účely rozpoznávání emocí, jelikož četnost neutrální emoce vysoce převažuje ostatní.

Celkem byly otestovány 4 detekční algoritmy. Nejlepší detekční schopnost vykazoval algoritmus MTCNN s F1 skóre rovno $0,913$ pokud $Iou \geq 0,5$. Při testování bez grafické karty však byla rychlost pouhých $0,877$ snímku za vteřinu. To je přibližně $6\times$ pomalejší než algoritmus Viola-Jones dosahující stále solidního F1 skóre rovno $0,823$. Segmentace prahováním, přestože je splněna podmínka výskytu pouze 1 osoby na fotografii dosahuje úspěšnosti velmi nízké.

V úloze identifikace osoby byla nejúspěšnější metodou DML s úspěšností $96,9\%$. Jedná se však obdobně jako u metody MTCNN o konvoluční neuronovou síť, proto trénování a testování na CPU je časově mnohem náročnější. Na rozdíl od metody MACE a PCA není vyžadována změna velikosti snímku na jednotnou velikost. Metoda PCA z důvodu velké variability dat selhává s celkovou úspěšností $12,8\%$ při volbě 50 hlavních komponent.

Poslední analyzovanou úlohou bylo rozpoznávání emocí, věku a pohlaví. Metoda YOLOv7 vykazuje téměř ideální tvar F1 a PR křivek v úloze rozpoznání pohlaví s hodnotou $mAP0.5$ rovno $0,961$. Rozpoznávání věku s celkem 7 klasifikačními třídami dosahuje solidní hodnoty $mAP0.5 = 0,543$, avšak při vyšších hodnotách confidence velmi strmě klesá F1 skóre. Úloha rozpoznávání emocí z důvodu nízkého zastoupení tříd smutek a naštvanost selhává, proto nebyla využita při testování návrhů vlastních neuronových sítí. Bylo prokázáno, že pro rozpoznávání pohlaví je možné využít i MLP síť s jednou skrytou vrstvou s úspěšností $81,66\%$. Nejúspěšnější sítí se stala rozšířená VGG7BN s úspěšností $89,58\%$. Naopak, při trénování úlohy rozpoznávání věku, vykazuje neuronová síť MLP a dvouvrstvá CNN chování náhodného klasifikátoru. Nejvyšší úspěšností $34,35\%$ dosáhla taktéž VGG7BN. Natrénování této úlohy s vyšší úspěšností by vyžadovalo komplexnější architekturu sítě.

Závěrem lze říci, že přístupy založené na konvolučních neuronových sítích vykazují vyšší úspěšnosti než metody tradiční. Avšak při absenci grafické karty, vyžadují mnohem větší časovou náročnost, to není vždy možné pro nasazení v aplikacích

vyžadující minimální latenci. S rychlostí rozvoje výpočetních technologií umožňujících komplexnější architektury neuronových sítí lze předpokládat směřování detekce a identifikace osob právě tímto směrem.

Literatura

1. Václav Hlaváč, M. S. *Zpracování signálů a obrazů* 2. vyd., 255. ISBN: 80-01-03110-1 (Vydavatelství ČVUT, Praha, 2005).
2. Adakane, D. *What are Haar features used in face detection* lis. 2019. https://medium.com/analytics-vidhya/what-is-haar-features-used-in-face-detection-a7e531c8332b.
3. Freund, Y. & Schapire, R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences* **55,** 119–139. ISSN: 0022-0000. https://www.sciencedirect.com/science/article/pii/S002200009791504X (1997).
4. Viola, P. & Jones, M. *Rapid object detection using a boosted cascade of simple features* in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* **1** (2001), I–I.
5. Mallick, S. *Histogram of oriented gradients explained using opencv* lis. 2021. https://learnopencv.com/histogram-of-oriented-gradients/.
6. Dalal, N. & Triggs, B. *Histograms of oriented gradients for human detection* in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* **1** (2005), 886–893 vol. 1.
7. Zhang, K., Zhang, Z., Li, Z. & Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters* **23,** 1499–1503. https://doi.org/10.1109%2Flsp.2016.2603342 (srp. 2016).
8. K, S. *Non-maximum suppression (NMS)* dub. 2021. https://towardsdatascience.com/non-maximum-suppression-nms-93ce178e177c.
9. Turk, M. & Pentland, A. *Face recognition using eigenfaces* in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1991), 586–591.
10. Savvides, M., Kumar, K. V. & Khosla, P. K. *Face Verification using Correlation Filters* in (2002).
11. Liu, W. *et al. SphereFace: Deep Hypersphere Embedding for Face Recognition* 2018. arXiv: 1704.08063 [cs.CV].
12. He, K., Zhang, X., Ren, S. & Sun, J. *Deep Residual Learning for Image Recognition* 2015. arXiv: 1512.03385 [cs.CV].

13. Schroff, F., Kalenichenko, D. & Philbin, J. *FaceNet: A unified embedding for face recognition and clustering* in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, čvn. 2015). https://doi.org/10.1109%2Fcvpr.2015.7298682.

14. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. *You Only Look Once: Unified, Real-Time Object Detection* 2016. arXiv: 1506.02640 [cs.CV].

15. Anwar, A. *What is average precision in object detection & localization algorithms and how to calculate it?* květ. 2022. https://towardsdatascience.com/what-is-average-precision-in-object-detection-localization-algorithms-and-how-to-calculate-it-3f330efe697b.