VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
DEPARTMENT OF INTELLIGENT SYSTEMS

ALGORITMICKÉ ŘEŠENÍ STANOVENÍ VĚKU OSOBY NA ZÁKLADĚ 2D FOTOGRAFIE ZA VYUŽITÍ UMĚLÉ INTELIGENCE

ALGORITHMIC SOLUTION FOR DETERMINING THE AGE OF A PERSON BASED ON 2D PHOTOGRAPHY USING ARTIFICIAL INTELLIGENCE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **PAVEL BEDNÁŘ**
AUTHOR

VEDOUCÍ PRÁCE **prof. Ing., Dipl.-Ing. MARTIN DRAHANSKÝ, Ph.D.**
SUPERVISOR

BRNO 2021

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav inteligentních systémů (UITS) Akademický rok 2020/2021

Zadání bakalářské práce

Student: **Bednář Pavel**
Program: Informační technologie
Název: **Algoritmické řešení stanovení věku osoby na základě 2D fotografie za využití umělé inteligence**
Algorithmic Solution for Determining the Age of a Person Based on 2D Photography Using Artificial Intelligence
Kategorie: Zpracování obrazu

Zadání:

1. Prostudujte literaturu týkající se stanovení věku člověka z 2D obrazu, příp. přidruženou antropologickou či medicínskou literaturu.
2. Připravte vhodný dataset pro učení a testování neuronové sítě agregací nějakého veřejně dostupného datasetu a databáze Ústavu antropologie PřF MU.
3. Navrhněte algoritmické řešení, které využije natrénovanou neuronovou síť ke stanovení věku člověka a zároveň využije expertní systém, který bude založen na významných rysech obličeje (např. různé typy vrásek).
4. Navržený algoritmický postup z bodu 3 implementujte.
5. Otestujte úspěšnost systému a diskutujte možná rozšíření.

Literatura:

- ROTHE, Rasmus; TIMOFTE, Radu; VAN GOOL, Luc. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 2018, 126.2-4: 144-157.
- LIU, Xinhua, et al. Face Image Age Estimation Based on Data Augmentation and Lightweight Convolutional Neural Network. *Symmetry*, 2020, 12.1: 146.
- SCHMELING, Andreas, et al. Forensic age estimation: methods, certainty, and the law. *Deutsches Ärzteblatt International*, 2016, 113.4: 44.
- PAN, Hongyu, et al. Mean-variance loss for deep age estimation from a face. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018. p. 5285-5294.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Drahanský Martin, prof. Ing., Dipl.-Ing., Ph.D.**
Konzultant: Urbanová Petra, doc. RNDr., Ph.D., MUNI
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1. listopadu 2020
Datum odevzdání: 12. května 2021
Datum schválení: 11. listopadu 2020

Abstrakt

Automatizované určení věku člověka z fotky obličeje představuje jednu z výzev v oblasti umělé inteligence a strojového učení. Určit věk, je i pro člověka mnohdy netriviální záležitost, narozdíl od jiných biologických charakteristik, jako je určení pohlaví nebo rasové příslušnosti. Přitom informace o věku jedince je pro určité situace velmi podstatná. Například při spáchání nějakého přestupku či trestného činu o výši trestu spolurozhoduje právě věk. Dále tuto informaci lze využít při analýze zákazníků komerčního subjektu a následnému přizpůsobení nabídky. Cílem této práce je tedy umět z fotografie lidského obličeje extrahovat jeho věk. Algoritmus se skládá ze dvou modulů. Pokud první modul řekne, že je osoba mladší 14 let, půjde obrázek ještě do druhého modulu. Dále je představena ještě jedna verze algoritmu s přidaným modulem zaměřeným na vybrané obličejové rysy. Ve všech modulech se nad obrázkem provedou transformace, jejichž výsledky se zprůměrují. Na závěr je algoritmus vyhodnocen na standardních datasetech pro určení věku a porovnán s aktuálně používanými metodami v této oblasti.

Abstract

Automated person's age estimation from a facial image is one of the challenges in the field of artificial intelligence and machine learning. Age estimation is often a non-trivial complexity for a person, unlike other biological characteristics such as determining gender or race. Information about an individual's age is very important for certain situations. For example, when committing an offense or crime, the amount of the sentence is co-determined by age. This information can also be used in the analysis of customers of a commercial entity and the subsequent adjustment of the offer. The aim of this work is to be able to extract his age from a photograph of a human face. The algorithm consists of two modules. If the first module says that the person is under 14 years old, the image will go to the second module. Furthermore, another version of the algorithm is proposed with an added module focused on selected facial features. In all modules transformations are performed on the image and their results are averaged. Finally, the algorithm is evaluated on standard datasets for age estimation and compared with state-of-the-art methods in this area.

Klíčová slova

hluboké učení, zpracování obrazu, konvoluční neuronové sítě, určení věku, Python

Keywords

deep learning, image processing, convolutional neural networks, age estimation, Python

Citace

BEDNÁŘ, Pavel. *Algoritmické řešení stanovení věku osoby na základě 2D fotografie za využití umělé inteligence*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Ing., Dipl.-Ing. Martin Drahanský, Ph.D.

Algoritmické řešení stanovení věku osoby na základě 2D fotografie za využití umělé inteligence

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana prof. Ing., Dipl.-Ing. Martina Drahanského, Ph.D. Další informace mi poskytla doc. RNDr. Petra Urbanová, Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Pavel Bednář
9. května 2021

Poděkování

Chtěl bych poděkovat vedoucímu mé práce prof. Ing. Dipl.-Ing. Martinovi Drahanskému, Ph.D. za toto téma, cenné rady a supervizování této práce. Dále bych chtěl poděkovat mé konzultantce doc. RNDr. Petře Urbanové, Ph.D. za poskytnuté antropologické informace a obličejovou databázi. Děkuji také Ing. Tomášovi Goldmannovi za jeho generátor 2D dat. V neposlední řadě děkuji MetaCentru za poskytnutí výpočetních zdrojů.

Obsah

Kapitola 1

Úvod

Obličej je plný biologických charakteristik, tím pádem existuje mnoho informací, které pomocí něj můžeme o daném člověku získat. Mezi hlavní patří určení věku, pohlaví, etnické příslušnosti nebo obličejového výrazu. Díky masivnímu rozvoji strojového učení a umělé inteligence v posledních letech vzniká spousta systémů, které dokáží takovéto informace získávat.

Během ontogeneze jedince přibývají na jeho obličeji různé typy vrásek, které se dále rozlišují podle stupně jejich hloubky. Mohlo by se zdát, že vrásky jsou záležitostí pouze starších a starých lidí, ale ve skutečnosti se první vrásky na obličeji objevují již okolo dvacátého roku života. S postupem času se vyskytují nové vrásky a ty staré se prohlubují [26]. Nicméně i přesto je proces stárnutí velmi individuální věc, která se na každém člověku projevuje trochu jinak a závisí na spoustě faktorů, jako třeba životní styl nebo prostředí, kde osoba žije.

Stanovení věku člověka z fotografie nebo videa má širokou škálu aplikací. Například v soudnictví, kde o výši trestu spolurozhoduje i fakt, jestli trestaný patří do věkové skupiny dítě, mladistvý, nebo dospělý. Poslední dobou, se zvyšující se migrační vlnou, přibývá případů, kdy je věk podezřelého neznámý, a to z důvodu, že nevlastní žádné průkazy totožnosti. Většinou se jedná o nelegální imigranty. V roce 2014 bylo vydáno v Berlíně 157 požadavků na určení věku imigranta, což je dvakrát více než počet z roku 2004 [42]. Dále policejní kamery, které by mohly automaticky detekovat věk, který v určitých případech může rozhodnout, jestli se jedná o přestupek, jako je třeba užívání alkoholu mladistvými. Taktéž získáním informace o věku lze blíže specifikovat pachatele nějaké trestné činnosti. Firmy či instituce jako je muzeum, zoologická zahrada, obchodní centrum apod. by mohly na základě automatizovaných dat o věku svých zákazníků přizpůsobovat svoji nabídku. Využití lze najít také na sociálních sítích. Cíle této práce jsou:

- Vytvořit dataset vhodný na trénování a testování tohoto systému použitím nějakého veřejně dostupného datasetu a vytvořením vhodného datasetu z 3D modelů Ústavu antropologie PřF MU.
- Navrhnout a implementovat algoritmus, který bude realizovat stanovení věku člověka z fotografie jeho obličeje s důrazem na využití významných obličejových rysů.
- Otestovat toto řešení, zhodnotit jeho funkčnost, porovnat ho s aktuálně používanými metodami a diskutovat možná rozšíření.

Struktura této práce je následující. Kapitola 2 se věnuje popisu lidského obličeje, způsobům snímání 2D dat z 3D modelů a shrnutí aktuálně používaných metod v oblasti určování věku

z obrazu. Kapitola 3 představuje návrh a implementaci navrhovaného algoritmu. V kapitole 4 je popsáno trénování, navrhovaný algoritmus je porovnán s konkurenčními algoritmy a jsou diskutována možná rozšíření. V kapitole 5 je pak celkové zhodnocení.

Kapitola 2

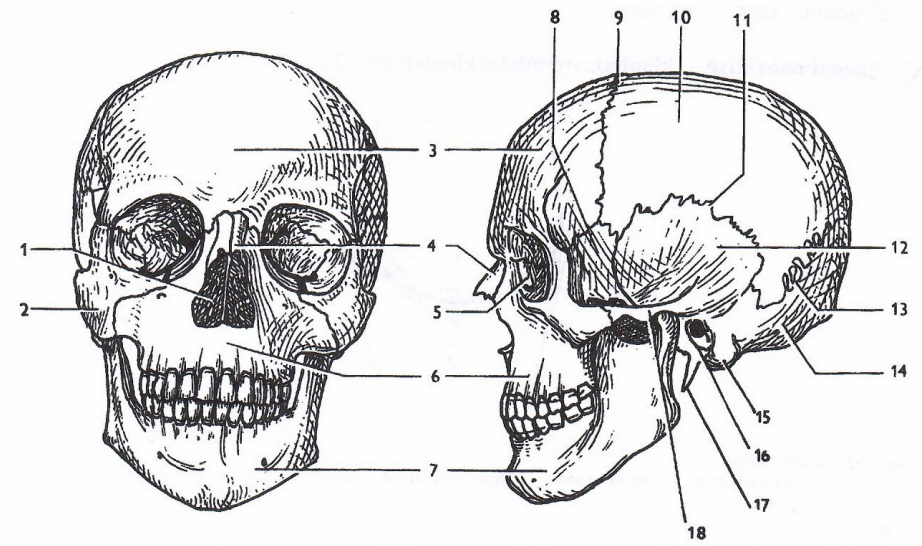
Současný stav

Na začátku této kapitoly budou popsána biologická fakta o lidském obličeji. Poté bude předveden způsob generování 2D dat z 3D modelů. Hlavní část této kapitoly je věnována vysvětlení jednotlivých fází procesu určení věku člověka, kterými jsou: předzpracování dat, extrakce parametrů, určení věku a aktualizace parametrů.

2.1 Anatomie obličeje

Z anatomie člověka jsem vybral relevantní témata pro tuto práci, což jsou popis lebky, obličejových svalů a vrásek.

2.1.1 Kostra lebky



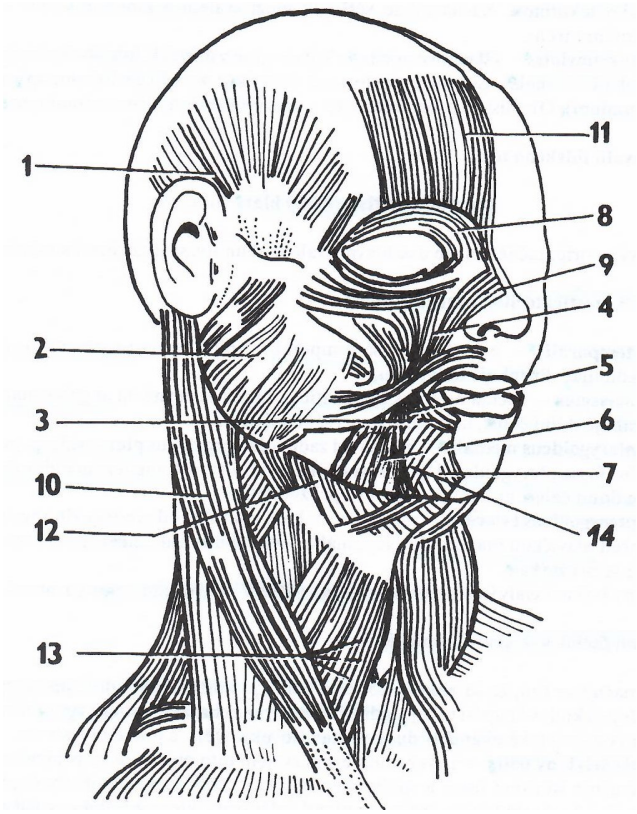
Obrázek 2.1: Kostra lebky [25]. 1 – *cavum nasi*, 2 – *os zygomaticum*, 3 – *os frontale*, 4 – *os nasale*, 5 – *os lacrimale*, 6 – *maxilla*, 7 – *mandibula*, 8 – *ala major ossis sphenoidalis*, 9 – *sutura coronalis*, 10 – *os parietale*, 11 – *sutura squamosa*, 12 – *squama temporalis*, 13 – *sutura lambdoidea*, 14 – *os occipitale*, 15 – *processus mastoideus ossis temporalis*, 16 – *pars tympanica*, 17 – *processus styloideus*, 18 – *pons zygomaticus*.

Kostru lebky (obrázek 2.1) dělíme na mozkovou (*neurocranium*) a obličejovou část (*splanchnocranium*). Tyto dvě části jsou od sebe odděleny hranicí, která se nachází zhruba 1 cm nad kořenem nosu, odtud podél oblouků nadočnicových k zevnímu zvukovodu až k hrbolu na týlní kosti (*protuberantia occipitalis externa*). Lebka je tvořena z mnoha kostí, které jsou spojeny pomocí švů.

Mozková část se dělí na vyklenutou klenbu lební (*calva*) a spodinu lební (*basis cranii*). První zmíněná část slouží jako schránka pro mozek, v druhé se pak nachází důležité otvory pro výstup hlavových nervů a vstup některých důležitých cév a míchy. Celá mozková část se skládá z týlní kosti (*os occipitale*), klínové kosti (*os sphenoidale*), dírkované ploténky kosti čichové (*lamina cribrosa ossis ethmoidalis*), čelní kosti (*os frontale*), spánkové kosti (*os temporale*), temenní kosti (*os parietale*), slzné kosti (*os lacrimale*), nosní kosti (*os nasale*) a radličné kosti (*vomer*).

Obličejová část se pak sestává z horní čelisti (*maxilla*), lícní kosti (*os zygomaticum*), patrové kosti (*os palatinum*), čichové kosti (*os ethmoidale*), dolní nosní skořepy (*concha nasalis inferior*), dolní čelisti (*mandibula*) a jazylky (*os hyoideum*). Mezi kosti obličejové části lebky řadíme rovněž sluchové kůstky (*ossicula auditus*) [25].

2.1.2 Svaly hlavy

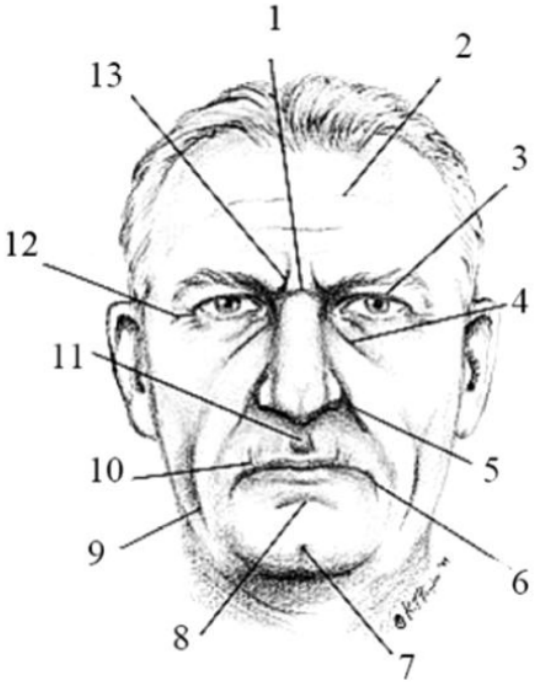


Obrázek 2.2: Svaly hlavy a krku [25]. 1 – *m. temporalis*, 2 – *m. masseter*, 3 – *m. orbicularis oris*, 4 – *m. levator labii sup.*, 5 – *m. buccinator*, 6 – *m. depresor anguli oris*, 7 – *m. triangularis*, 8 – *m. orbicularis oculi*, 9 – *m. nasalis*, 10 – *m. sternocleidomastoideus*, 11 – *m. frontalis*, 12 – *mm. suprahyoidei*, 13 – *mm. infrahyoidei*, 14 – *m. depressor labii inferioris*.

Svaly hlavy (obrázek 2.2) dělíme na dvě hlavní svalové skupiny, kterými jsou svaly žvýkací a mimické. Všechny žvýkací svaly jsou inervovány z třetí větve V. hlavového nervu (*n. mandibularis*). Mezi žvýkací svaly patří *m. temporalis*, *m. masseter*, *m. pterygoideus medialis* a *m. pterygoideus lateralis*. Typickou vlastností mimických svalů je, že jeden konec je fixován na kosti lebky a druhý se upíná do kůže. Dokáží vyjádřit okamžitý duševní stav jedince. Výraz a rysy konkrétního obličeje určuje klidové napětí těchto svalů. Podle oblasti výskytu se dělí na svaly štěrbiny ústní, svaly v oblasti očních víček, svaly v oblasti nosu, svaly klenby lební, svaly boltce ušního a *m. buccinator*. Tyto svaly jsou inervovány ze VII. hlavového nervu (*n. facialis*) [25].

2.1.3 Vrásky

Vrásky v obličeji jsou velmi individuální znak, který je ovlivněn spoustou faktorů jako je např. životní styl nebo prostředí kde člověk žije. Vznikají s přibývajícím věkem, kdy dochází ke ztenčování škáry a ubývání množství papil, které jsou vysílány k epidermis. Mezi těmito vrstvami dochází ke ztrátě pevnosti, což vede ke zmíněné tvorbě vrásek. Ty se tvoří kolmo k níže přítomným svalovým vláknům. Do 20 let věku se objevují pouze náznaky rýh, které jsou dány genetickým podkladem. Po 20. roku života se již vytvoří první vrásky v oblasti glabely a v okolí očí, způsobené obličejovou mimikou. Od 30 let přibydou nové rýhy a stávající se dále prohlubují (např. nasolabiální rýha). Po 40. roku vznikají vrásky kolem uší a mohou se tvořit i váčky pod očima. Stávající vrásky opět pokračují v prohlubování. Přibližně v 60 letech přibývají vrásky kolem horního rtu a postupně dochází k vrásčitosti celé tváře [28]. Přehled obličejových vrásek je na obrázku 2.3.



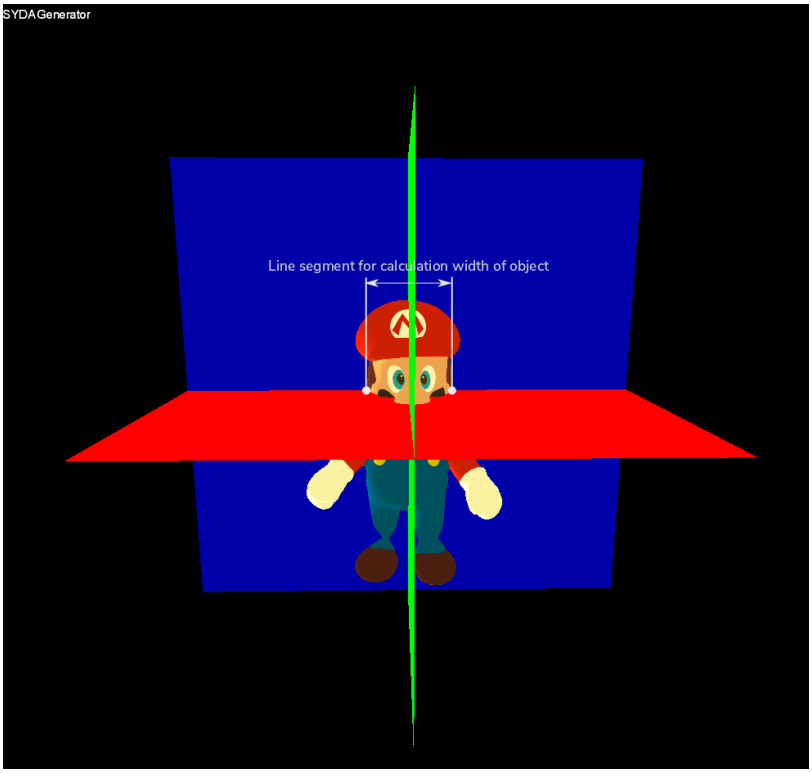
Obrázek 2.3: Vrásky a linie obličeje [28]. 1 – Horizontální nosní linie, 2 – Horizontální čelní linie, 3 – Horní oční vrásky, 4 – Dolní oční vrásky, 5 – Nasolabiální rýha, 6 – Vrásky ústního koutku, 7 – Jamka v oblasti brady, 8 – Mentolabiální rýha, 9 – Bukomandibulární vrásky, 10 – Svislé vrásky na horním rtu, 11 – Vrásky v oblasti filtra.

2.2 Způsoby snímání 2D dat z 3D modelů

Za účelem zpřesnění klasifikace výsledného algoritmu chci vytvořit trénovací dataset z databáze 3D modelů obličejů Fidentis [47]. Kvalitní a široký dataset je totiž jedna z klíčových věcí při trénování neuronové sítě. Je potřeba tedy nasnímat 2D data z 3D modelů.

2.2.1 SyDa Generátor

Jedná se o software pro generování 2D datasetů z 3D modelů. Program funguje tak, že generuje snímky pomocí projekce 3D modelu do přesně stanovených pozic vůči zvolenému pozadí. Obsahuje široké možnosti nastavení generovacích parametrů, jako třeba rozsah rotace, krok rotace, pozici světla, bloom efekt atp. Pomocí skriptování lze dosáhnout plně automatického generování [19]. Na obrázku 2.4 je ukázka 3D modelu otevřeném v SyDa Generátoru[1].



Obrázek 2.4: Ukázka prostředí SyDa Generátoru [19].

2.3 Předzpracování dat

Předzpracování dat je soubor technik, které mohou významně vylepšit výslednou úspěšnost modelu. Může se jednat např. o otočení obrázku tak, aby oči člověka ležely na jedné horizontální přímce, přiblížení obrázku, odstranění pozadí za obličejem nebo, pomocí různých operací, obrázky transformovat a vytvořit tím větší množinu trénovacích dat, což zvyšuje pravděpodobnost zabránění přetrénování a získání lepší obecnosti. Ideální vstupní obrázky

[1] www.fit.vut.cz/research/product/668/

by tedy měly mít stejnou velikost, měly by obsahovat minimum pozadí, obličej v nich by měl být vycentrovaný a v přirozené poloze, tzn. nenahnutý doleva či doprava. Velmi důležitý, pro tuto část, je kvalitní obličejový detektor [45].

2.3.1 Afinní transformace

Tato metoda se objevuje v [14]. Zde je využit speciální detektor 68 obličejových rysů, jako jsou např. ústa, nos, kraje očí apod. Výběrem ideálních souřadnic těchto rysů získáme nějakou afinní transformaci, která zajistí, že obličej bude správně zarovnán, avšak chyby v detekování nějakého rysu mohou vyústit v nestabilní výsledky transformace. Některé obličejové charakteristiky (např. oči) jsou snadněji a s větší dávkou jistoty lokalizovatelné než jiné (např. lícní kosti). Pro zohlednění míry jistoty lokalizace různých částí obličeje je potřebné vědět míru přesnosti určení každého z 68 rysů. To je nám ale známé až po dokončení zarovnání obličeje. Na vyřešení tohoto problému obousměrné závislosti se používá IRLS algoritmus. Obrázek 2.5 demonstruje výše zmíněné principy.

Obrázek 2.5: Vizualizace metody. Vlevo jsou ideální souřadnice obličejových rysů. Uprostřed jsou rysy detekované na reálném obličeji. Vpravo je stupnice jistoty lokalizace [14].

2.3.2 Rotace pomocí detekce významných rysů obličeje

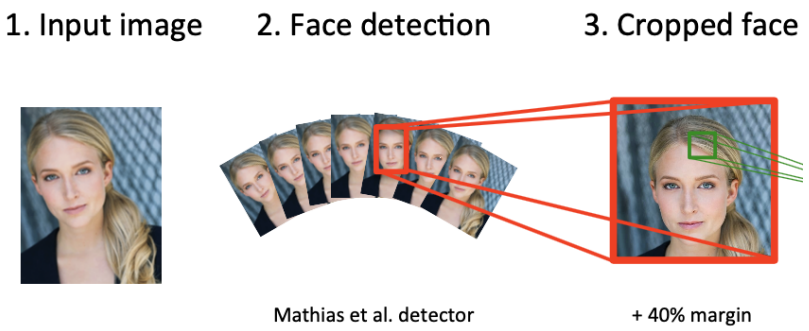
Tato technika bývá používána velmi často. Proces začíná u detektoru obličejových rysů, který nalezne obličej a jeho hlavní charakteristiky, ten se ořízne a natočí tak, aby mezi oběma očima byla horizontální přímka. Obrázek 2.6 ukazuje průběh metody.

Obrázek 2.6: Vizualice předzpracování oříznutím a rotací [32].

Výše zmíněné metody se mohou i kombinovat [3]. Podle [30] jsou metody rotace obecně účinější než metody pomocí afinní transformace. Úspěšnost věkové klasifikace byla testována na modelech jako jsou CaffeNet [27], GoogLeNet [46] apod. Tento závěr pravděpodobně dokazuje i fakt, že nejúspěšnější metody jsou založené na preprocessingu pomocí rotace.

2.3.3 Rotace pomocí skóre obličejového klasifikátoru

Jedná se o modifikovanou verzi výše zmíněného přístupu. [41] vezme vstupní obrázek a vytvoří množinu, která obsahuje několik těchto různě natočených obrázků. Následně tuto množinu předhodí obličejovému detektoru [34], který vybere obrázek s největším skóre. Část detekovaná jako obličej se z vybraného obrázku ořízne, přidá se k ní 40 % z původního obrázku vybraného obličejovým detektorem (pokud původní obrázek není dostatečně velký bude se opakovat poslední známý pixel) a poměrově se zmenší tak, aby výsledná velikost byla 256×256 pixelů. Tento proces zajistí, že všechny obrázky budou obsahovat stejný poměr obličeje a pozadí, budou stejně velké a obličej bude správně zarovnaný. Průběh je vidět na obrázku 2.7. Podle závěrů této práce, zarovnávání obrázku podle skóre obličejového klasifikátoru vede k zhruba 5krát méně chybným výsledkům než klasičtější přístup zarovnávání podle detekce významných rysů obličeje. Také autoři dospěli k tomu, že přidání nějakého pozadí k obličeji zvyšuje přesnost v dalších fázích procesu určení věku.



Obrázek 2.7: Jednotlivé fáze vylepšené metody předzpracování pomocí rotování [41].

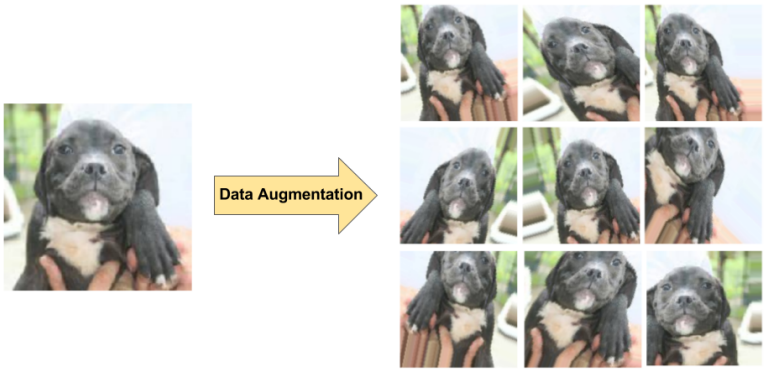
2.3.4 Zvětšení množiny trénovacích dat

Používá se pro předejití přetrénování, získání přesnějších výsledků klasifikace nebo při klasifikaci do více tříd, k rovnoměrnosti počtu dat ve třídách. Tato činnost může probíhat na dvou místech. Buď v prostoru vstupních dat (obrázků) nebo v prostoru příznaků [48]. To v architektuře konvolučních neuronových sítí odpovídá fázi předzpracování dat resp. extrakce parametrů.

Prostor vstupních dat

Tento přístup je obecně používanější a podle [48] i mírně účinější. Lze to také vnímat jako způsob předzpracování dat, který zlepšuje generalizaci modelu. V testovací fázi se může použít obrázek, který se rozmnoží tak, že se ořízne na vždy trochu jiných místech a výstup se určí z jednotlivých dílčích výsledků. Jiný způsob předzpracování může být třeba snížení variací v datasetu, které model musí zohlednit. Toto opět pomůže změnšit chybu generalizace a také nám dovolí použít jednodušší model s méně parametry. Jednoduché úlohy by měly využívat jednoduché modely, které ovšem mají tendenci lépe generalizovat. Ke snížení variability v datasetu bychom měli přistoupit, pokud danou přebytečnou variabilitu dokážeme jasně popsat a jsme si jistí, že nemá vliv na danou úlohu. Pokud pracujeme s velkými modely a datasety, je lepší toto manuálně nedělat a nechat model ať se sám rozhodne co je a co není pro něj podstatné.

Jak již bylo naznačeno, tak pokud máme k dispozici malý objem trénovacích dat, pak tyto techniky dokáží vylepšit schopnost získání větší obecnosti modelu. Tento přístup se více využívá u klasifikace, protože vstupní obrázky patří k jedné seperátní třídě. Stačí tedy aplikovat vybranou transformaci a vznikne nové dato, které přiřadíme ke stejné třídě. Důležité je aplikovat tranformace, které nezmění třídu. Například pokud klasifikujeme čísla, tak nedává smysl rotovat vstupní data o 180°, protože dojde k záměně čísel 6 a 9. K rozšíření datasetu se používají operace jako je náhodná translace, rotace, přiblížení a horizontální či vertikální převrácení [20]. Ukázka je na obrázku 2.8.



Obrázek 2.8: Ukázka augmentace dat [2].

Prostor příznaků

Zvětšení množiny trénovacích dat v prostoru příznaků může být výhodnější, protože ve výše uvedeném přístupu, po aplikování afinních transformací, se vstupní obrázek může změnit tak, že již nebude odpovídat původní věkové anotaci. Synthetic Minority Over-Sampling Technique (SMOTE) [11] v N-rozměrném prostoru všech dat jedné třídy náhodně vybere několik bodů, proloží jimi přímku a na ní náhodně určí jeden bod, který prohlásí za nové dato. Takto se dá vytvořit libovolný počet dat jakékoliv třídy, za předpokladu, že daná třída již obsahuje nějaký počet trénovacích dat. Výhoda této metody je, že je zcela nezávislá na úloze, a to z důvodu, že pracuje až s příznaky. Tento algoritmus se často používá na dorovnání počtu dat ve všech třídách. Z této metody vznikají její novější deriváty jako je např. Density-Based Synthetic Minority Over-Sampling Technique (DBSMOTE) [6]. Zde se najde shluk všech dat jedné třídy v N-rozměrném prostoru, určí se jeho střed a nová data se generují náhodně v jeho okolí, které je ohraničeno vzdáleností od středu shluku. Nevýhoda je, že nová data se generují v blízkém okolí středu shluku, a tím pádem může snadno dojít k přetrénování.

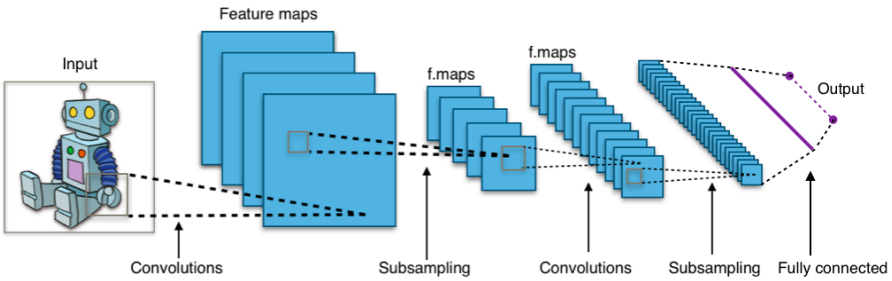
2.4 Extrakce parametrů

Tato vrstva se využívá k získání nějakých parametrů z obrázku obličeje na základě kterých pak určitou technikou vytvoříme věkovou predikci. Zpočátku se v této oblasti začaly používat BIF, neboli příznaky inspirované biologickými rysy. Např. [21] používá pyramidu Gaborových filtrů, které hledají v konkrétní oblasti obrázku konkrétní frekvenční rozsah.

Ukázalo se ale, že přesnost takových metod není ideální a s rostoucím úspěchem CNN v oblastech počítačového vidění, jako je detekce objektů, klasifikace obrázků nebo rozpoznávání obličeje, se konvoluční neuronové sítě začaly využívat i ke stanovení věku. Nyní se CNN, v různých architekturách, používají v drtivé většině systémů na určení věku člověka.

2.4.1 Konvoluční neuronové sítě

Jedná se o speciální druh neuronových sítí určených ke zpracování dat s mřížkovitou topologií. Příkladem jsou data časových řad, která jsou interpretována jako 1D grid nebo obrázková data, která představují 2D grid skládající se z pixelů. Druhý případ užití je typičtější a sklízí velký úspěch v praktických aplikacích. Konvoluční neuronové sítě jsou, jak lze i z názvu odvodit, neuronové sítě obsahující alespoň jednu konvoluční vrstvu. Obsahují také nějaký počet podvzorkovacích (pooling) vrstev a plně propojených (fully connected) vrstev [20]. Typicky se několikrát střídají kovoluční a pooling vrstvy a na konci je jedna nebo i více fully connected vrstev. Vstup může mít velikost např. 32×32×3, což v sobě kóduje hodnoty pixelů obrázku s rozlišením 32×32 pixelů ve 3 barevných RGB kanálech. Konvoluční vrstva obsahuje např. 12 různých filtrů o velikosti 5×5×3, které dokáží detekovat různé objekty v obrázku. Výstup této vrstvy by měl velikost 32×32×12. Následující pooling vrstva podvzorkuje obrázek na prostorových dimenzích (šířka, výška). Např. ze 4 sousedících pixelů zachová pouze ten s největší hodnotou, a tím obrázek 4krát zmenší na rozlišení 16×16×12. Fully connected vrstva propojí všechny neurony z předchozí vrstvy s neurony následující vrstvy, která bývá poslední a její velikost je určena počtem tříd do kterých chceme klasifikovat, tzn. pokud se rozhodneme klasifikovat číslice, pak 1×1×10 [18]. Architektura typické CNN je na obrázku 2.9.



Obrázek 2.9: Schéma typické konvoluční neuronové sítě [5].

Konvoluční vrstva

V tomto kontextu mluvíme o diskrétní konvoluci, protože naše vzorky, nebo-li pixely, jsou diskrétní veličina. Jedná se o matematickou komutativní operaci definovanou takto:

$$x[i] * h[i] = \sum_k x[k]h[i-k] \tag{2.1}$$

kde x je vstup, h je kernel a výsledek se často označuje jako feature maps. Výše zmíněný vztah (2.1) definuje 1D konvoluci, tedy pouze v jedné ose. V naší situaci, což je klasifikace obrázků, ale máme 2D grid, tudíž potřebujeme 2D konvoluci. Ta má následující předpis:

$$x[i,j] * h[i,j] = \sum_k \sum_l x[k,l]h[i-k, j-l] \tag{2.2}$$

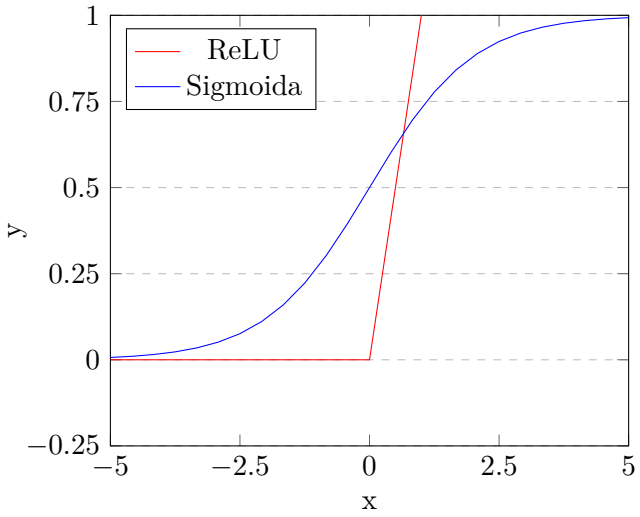
Výše uvedený vztah 2.2 by stačil pokud bychom měli v celé CNN jen jednokanálový vstup (např. černobílý obrázek) a 1 kernel, tudíž by v celé CNN figurovala pouze 1 feature map. Ve velké většině případů ale máme tříkanálový vstup (např. RGB obrázek) a více konvolučních filtrů. Konvoluce je v tomto případě definována takto:

$$conv(x,h)_{i,j} = \sum_k \sum_l \sum_m x[k,l,m]h[i-k, j-l, m] \tag{2.3}$$

kde m je dimenze počtu vstupů a konvoluční filtr se přes tuto dimenzi nepohybuje, pouze sčítá výsledky konvolucí jenotlivých vstupů. Počet výstupů (feature maps) je dán počtem konvolučních filtrů (kernels). Po každé konvoluční vrstvě následuje nelinearita, která je nutná, jinak by celá CNN dokázala vyprodukovat pouze nějakou lineární kombinaci vstupů. Nejčastěji používané nelinearity jsou ReLU (vzorec 2.4) nebo Sigmoida (vzorec 2.5). Jejich graf je znázorněn na obrázku 2.10. Přednost konvolučních filtrů je, že dokáží dobře identifikovat od různých typů hran (např. horizontální hrany, vertikální hrany) po konkrétní objekty (např. auto, pes) [8].

$$R(x) = max(0, x) \tag{2.4}$$

$$S(x) = \frac{1}{1+e^{-x}} \tag{2.5}$$



Obrázek 2.10: Graf dvou používaných aktivačních funkcí.

Důvod, proč se používá právě tato vrstva a její hlavní benefit je lokální propojení. Když pracujeme s vysokodimenzionálními vstupy jako jsou obrázky, je nepraktické propojit každý pixel s jedním neuronem, protože by to velmi zvýšilo počet parametrů sítě. Místo toho se tedy propojí skupina lokálních pixelů s jedním neuronem. Lokální propojení je vždy v prostorové rovině (šířka, výška) a je dáno velikostí konvolučního filtru. V poslední dimenzi (hloubkové) je velikost vždy konstantní a odpovídá počtu vstupních kanálů [18].

Pooling vrstva

Tato vrstva je určena ke snížení prostorových dimenzí obrázku, což vede k celkovému zmenšení počtu parameterů, a tím se sníží objem provedených výpočtů, ale také se redukuje

riziko přetrénování. Pooling vrstva pracuje nezávisle na jednotlivých hloubkových dimenzích a jejich počet neovlivňuje. Odehrává se v prostorové rovině, kde typicky na hodnoty z oblasti 2×2 aplikuje operaci max a tuto oblast nahradí výledkem max operace [18].

Fully connected vrstva

Jak název napovídá, vstupní a výstupní neurony této vrstvy jsou plně propojeny. Jedná se tedy o klasickou neuronovou síť. Obecně, konvoluční vrstva a fully connected vrstva, jsou na sebe vzájemně převoditelné. Rozdíl mezi nimi je, že výstupní neuron konvoluční vrstvy zastupuje region pixelů, zatímco pokud bychom použili na klasifikaci obrázků klasickou neuronovou síť, pak jeden neuron by zastupoval jeden pixel [18]. Abychom získali výslednou pravděpodobnost jednotlivých tříd, tak se na poslední vrstvu aplikuje funkce Softmax (rovnice 2.6), která výsledky znormalizuje.

$$S(x_i) = \frac{e^{x_i}}{\sum_{j=0}^{N} e^{x_j}} \tag{2.6}$$

kde x_i je třída, jejíž pravděpodobnost chceme znát a N je počet tříd do kterých klasifikujeme.

2.4.2 Vlastnosti konvolučních neuronových sítí

Vliv na kvalitu extrakce parametrů a následnou klasifikaci, má více faktorů. V [4] se zaměřili na tyto čtyři kritéria pro které hledají ideální hodnoty:

- Hloubka CNN
- Předtrénování
- Mono-task vs. multi-task způsob trénování
- Zakódování predikovaného věku

Hloubka CNN

Co se týče hloubky CNN, tak zjistili, že větší vliv má počet konvolučních vrstev než počet fully connected vrstev. Proto se v dalším zkoumání zaměřili na kovoluční vrstvy, kde zkusili architektury s 2, 4, 6, nebo 8 konvolučními vrstvami. Ukázalo se, že čím více vrstev, tím větší přesnost. Volbu hloubky CNN ovšem musíme přizpůsobit také počtu trénovacích dat, které máme, a to tak, aby nedošlo k přetrénování. Určování věku je tedy komplexní problém, narozdíl třeba od určení pohlaví osoby, které v tomto článku zkoumají také. U určení pohlaví se při větší hloubce CNN její přesnost jen lehce zvyšuje. Při použití 8 konvolučních vrstev se dokonce sníží, a to na úroveň použití 2 vrstev.

Předtrénování

Předtrénování na jiné úloze je další předmět zkoumání tohoto článku. Obecně předtrénování je užitečná věc, která zpřesní výsledný odhad. Rozsáhlých veřejně dostupných datasetů pro určení věku osoby není mnoho, což mohla být motivace pro zabývání se předtrénováním na jiné úloze než je stanovení věku člověka. Jako nejlepší se ukázalo předtrénování na úloze rozpoznání obličeje, kde dostupné datasety jsou násobně větší. Z důvodu, že rozpoznávání

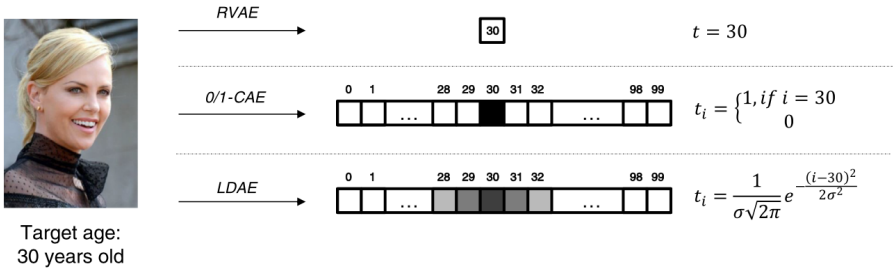
obličeje a určení věku nejsou až tolik podobné činnosti, nepředpokládal se výrazný vliv na výslednou predikci, ale nastalo alespoň malé zlepšení. Lze tedy říci, že z předtrénování na rozpoznávání obličeje si neuronová síť odnese nějakou informaci, která dokáže zlepšit klasifikaci i pro úlohu stanovení věku osoby.

Mono-task vs. multi-task způsob trénování

Multi-task trénování, nebo-li učení více úloh najednou, zde pohlaví a věk, výsledné přenosti nepomohlo. Důvod by mohl být ten, že síť již byla předtrénována na úloze rozpoznavání obličeje a rozpoznávání pohlaví může být vnímáno jako podmnožina nebo jedna z charakteristik rozpoznání obličeje, takže se síť nedozvěděla žádnou novou informaci.

Zakódování predikovaného věku

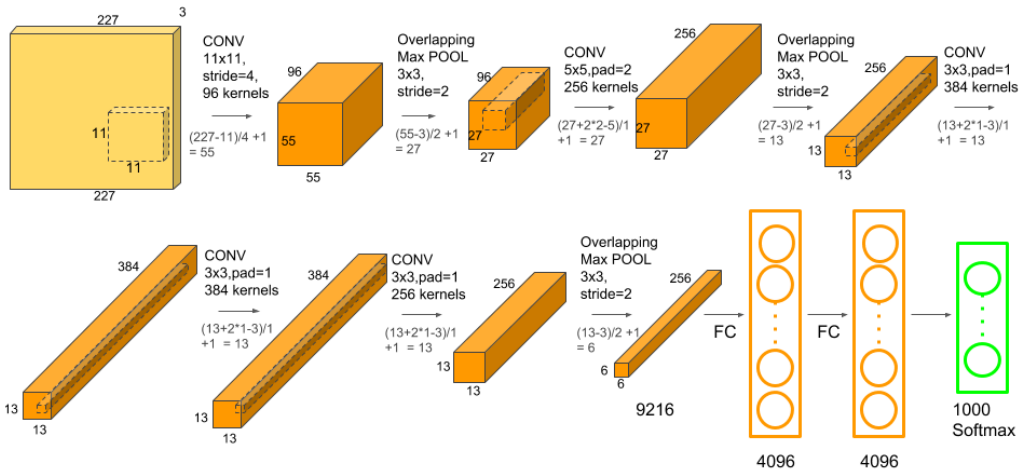
Vizualizaci zakódování věku ukazuje obrázek 2.11. První možnost je použití regrese, tedy jednoho reálného čísla. Zde značeno jako RVAE. Dále můžeme použít vektor s binárními hodnotami o velikosti všech věkových tříd (0/1-CAE), kde hodnota na indexu s reálným věkem osoby je 1 a ostaní jsou 0. Poslední možnost (LDAE) je vylepšení předchozího s tím rozdílem, že hodnoty vektoru nejsou binární, ale odpovídají gaussovskému rozložení pravděpodobnosti se středem v reálném věku osoby. Tato metoda obdržela nejlepší výsledky. Obsahuje to nejlepší z obou předchozích metod, a to ordinalitu z RVAE a nelineární modelování věku z 0/1-CAE.



Obrázek 2.11: 3 zkoumané způsoby zakódování věku [4].

2.4.3 AlexNet

Jednou z prvních prací, která popularizovala konvoluční neuronové sítě v počítačovém vidění, byl AlexNet [29]. AlexNet se účastnil ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 a výrazně překonal druhý nejlepší model (top 5 error 15,3 % vs. 26,2 %). Jeden z rozdílů oproti předchozím modelům je jeho velikost (počet parametrů), která je mnohem větší než v sítích používaných do té doby (např. LeNet [31] z roku 1998). Obsahuje přibližně 60 milionů parametrů, 5 konvolučních a 3 fully connected vrstvy. Architektura je ukázána na obrázku 2.12. V jedné konvoluční vrstvě je více kernelů o stejné velikosti. Např. v první konvoluční vrstvě je 96 kernelů o velikosti 11×11×3. Po druhé konvoluční vrstvě následují Overlapping Max Pooling vrstvy. Overlapping Max Pooling se od Max Poolingu liší tím, že kernel má velikost 3×3 se stride 2, takže se krajní pixely překrývají. Po páté konvoluční vrstvě následuje Overlapping Max Pooling vrstva, jejíž výstup jde do série dvou fully connected vrstev a následně je aplikována funkce Softmax [35].



Obrázek 2.12: Architektura AlexNet [35].

2.4.4 VGG-16

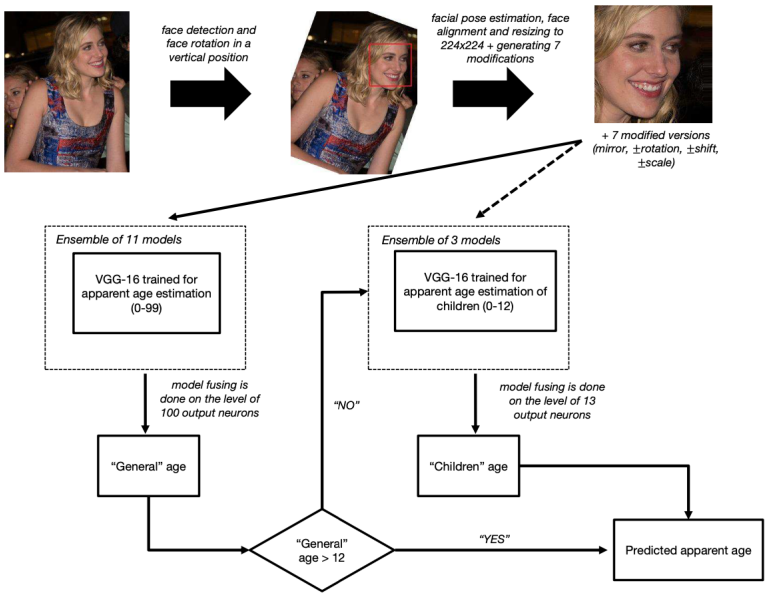
Jedná se o hlubokou konvoluční neuronovou síť s třinácti konvolučními a třemi fully connected vrstvami (tabulka 2.1) [44]. Poslední vrstva obsahuje 1000 neuronů. Tato CNN dosahuje velmi dobrých výsledků extrakce a jsou veřejně dostupné její předtrénované modely. Například v [41] použili již předtrénovanou síť na datasetu ImageNet [13] a dotrénovali ji na jimi představeném IMDB-WIKI datasetu.

Vrstva	Velikost výstupu	Počet výstupů
Vstupní obrázek	224×224	3
2× Konvoluční vrstva + ReLU	224×224	64
MaxPooling	112×112	128
2× Konvoluční vrstva + ReLU	112×112	128
MaxPooling	56×56	256
3× Konvoluční vrstva + ReLU	56×56	256
MaxPooling	28×28	512
3× Konvoluční vrstva + ReLU	28×28	512
MaxPooling	14×14	512
3× Konvoluční vrstva + ReLU	14×14	512
MaxPooling	7×7	512
2× Fully connected + ReLU	1×1	4096
Fully connected + ReLU	1×1	1000
Softmax	1×1	1000

Tabulka 2.1: Architektura VGG-16.

V [3] použili tuto neuronovou síť pro určení zdánlivého věku. Principy z této práce lze použít i pro naši úlohu, protože oni nejprve natrénovali model pro určení reálného věku a poté ho pouze dotrénovali na určování zdánlivého věku osoby. Algoritmus nejprve vezme

vstupní obrázek, několikrát ho natočí, všechny takto získané obrázky předhodí obličejovému klasifikátoru, a ten s největším skóre se vybere a ořízne se tak, že ke čtverci s obličejem se přidá ještě 40 % jeho okolí. Pokud obličejový detektor žádný obličej nenalezne, vstupní obrázek se přibliží a algoritmus začne znovu. Dále je na obrázek aplikována afinní transformace, abychom dosáhli co nejlepšího zarovnání a zmenší se na vstupní rozlišení sítě VGG-16, což je 224×224 pixelů. Na výsledný obrázek se aplikuje ještě 7 vstupních operací (jako např. malá rotace, zrcadlení, posunutí atp.). Takto získaných 8 obrázku je již kompletní vstup CNN. Samotná CNN je předtrénována na detekci obličeje, natrénována na IMDB-WIKI cleaned datasetu (jimi upravený IMDB-WIKI dataset) a je použito LDAE kódování věku z [4]. Pokud je výsledek klasifikace >12, jedná se již o konečnou predikci. Jinak je obrázek postoupen další CNN, která je stejná jako původní, ale navíc je dotrénována na jejich soukromém datasetu obsahujícím obrázky dětí. Využito je 0/1-CAE kódování věku, opět z [4], a to z důvodu, že u dětí je méně tříd a blízké věky si nejsou tolik podobné jako u dospělých, např. 30letý a 35letý jsou si podobnější něž 5letý a 10letý, ikdyž je mezi nimi stejný věkový rozdíl. Tato predikce už je finální. Schéma architektury je naznačeno na obrázku 2.13. V originální práci je ještě síť dotrénována na datasetu pro určení zdánlivého věku za použití Cross-Validation, protože zmíněný dataset není příliš velký. Tím pádem jim vznikne pro oba modely více CNN, jejíchž výsledky se zprůměrují.

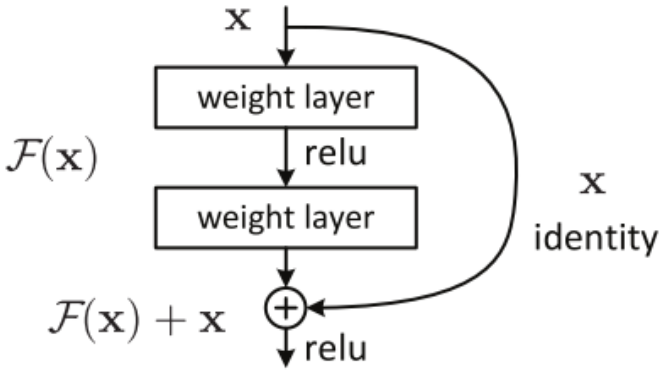


Obrázek 2.13: Testovací schéma daného systému [3].

2.4.5 ResNet

Residual network [24] je konvoluční neuronová síť čítající variace ResNet18, ResNet34, ResNet50, ResNet101 a ResNet152. Od té doby co AlexNet [29] získal vítězství v ILSVRC 2012, je Residual network jedna z neprůkopnějších prací v oblasti hlubokého učení a počítačového vidění. Jeho přednost spočívá ve velké hloubce, ale přitom působivé výpočetní rychlosti. Tento fakt pak umožňuje zvýšení výkonnosti aplikací počítačového vidění.

Obecně nyní existuje trend, že ke zvyšování přesnosti konvoluční neuronové sítě se musí jít cestou zvyšování její hloubky. Zhruba od roku 2012 jsou nejlepší CNN čím dál hlubší. Zatímco výše zmíněný AlexNet má pouze 5 konvolučních vrstev, tak např. VGG má již 16. Nicméně pouhé přidávání vrstev za sebe také nestačí. Při dosažení určité hloubky se začně projevovat Vanishing Gradient problém. Tento problém znamená, že když je gradient zpětně propagován do předchozích vrstev, tak opakovaný součin může zapříčinit, že je jeho hodnota velmi malá. Toto může vyústit v to, že chybová funkce přestane klesat, nebo se dokonce začne mírně zvyšovat. Nejdůležitější část ResNetu je představení tzv. Identity Shortcut Connection, což se využívá v Residual Blocku (obrázek 2.14). Myšlenka je, že pomocí těchto částí síť dokáže přeskočit jednu, či více vrstev neuronové sítě [15].

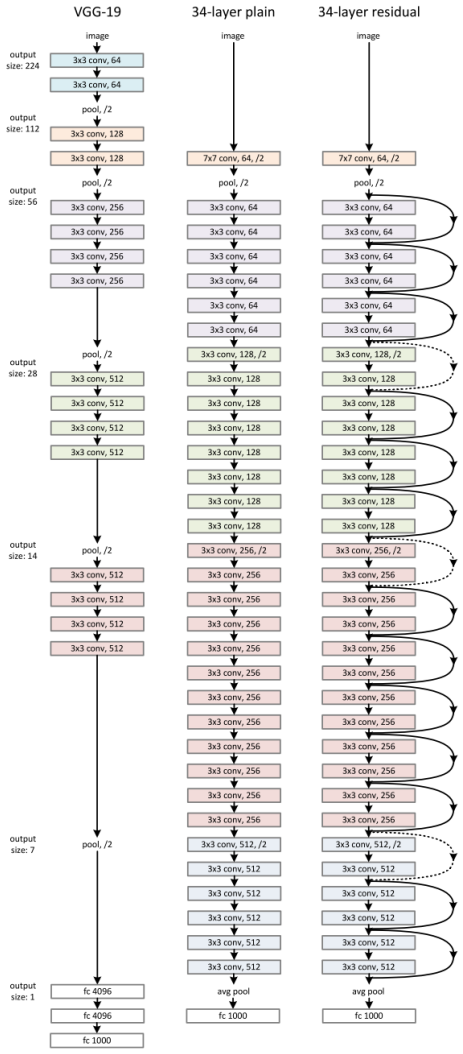


Obrázek 2.14: Residual Block z ResNet [15].

2.4.6 ShuffleNetV2

Jedná se o lehkou konvoluční neuronovou síť, myšleno ve smyslu, že obsahuje méně parametrů a trénuje se rychleji [33]. V [32] se používá modifikovaná verze ShuffleNetV2 s přídaným Mixed attention mechanismem. Originální ShuffleNetV2 je složen ze 7 vrstev uvedených v tabulce 2.2. Počet výstupů lze změnit. Zde je využit 2krát mód, ale existují varianty 0,5, 1, 1,5, které počet výstupních feature maps sníží, ovšem zároveň se sníží i kvalita extrahování parametrů. Modifikace architektury použitá v této práci spočívá v přizpůsobení neuronové sítě k účelům určování věku, tzn. počet výstupních neuronů byl snížen na 101 a také za tuto poslední fully connected vrstvu byla přidána regresní vrstva o jednom neuronu. Dále do Basic unit byl přidán Mixed attention mechanismus. Rozdíl mezi Basic unit z ShuffleNetV2 a této práce je ukázán na obrázku 2.16. Zde bude popsána rozšířená verze s Mixed attention modulem.

Nejprve se feature maps rozdělí do dvou větví a spojí se až v konkatenačním modulu. Pravá vetěv se ještě rozdělí na první a druhou část, které se spojí na konci Mixed attention modulu. První část projde třemi konvolučními vrstvami a poté vstoupí do Mixed attention modulu. Ten sestává z Channel attention modulu a Spatial attention modulu. V Channel attention modulu feature maps projdou konvolucemi a vznikne jich nový počet. Každá tato 2D feature map se transformuje na jedno reálné číslo, to se naváhuje dle určitého parametru, který modeluje korelaci mezi jednotlivými feature maps, vznikne vektor vah, kterým se vynásobí 2D feature maps po transformaci a tímto procesem se upraví důležitost jednotlivých feature maps. Spatial attention modul vezme všechny feature maps a provede mean a max v jejich dimenzích, nebo-li vezme si např. první příznak, podívá se na všechny

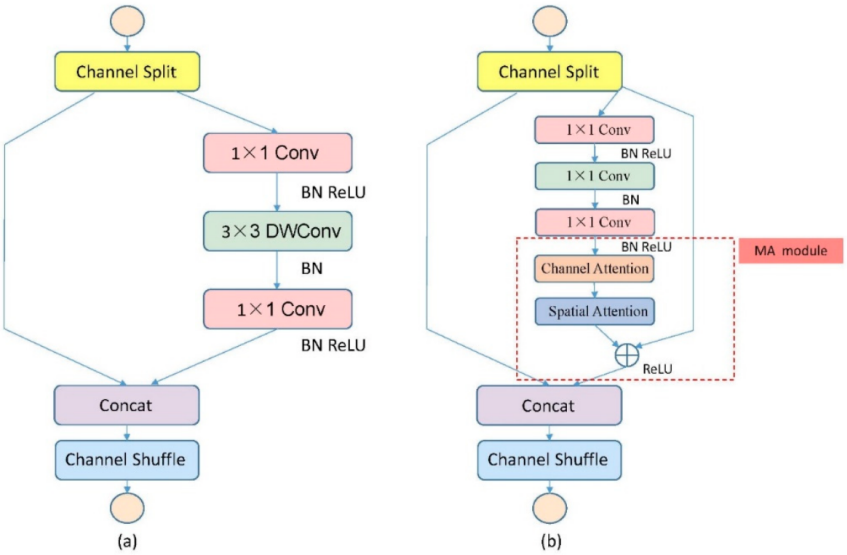


Obrázek 2.15: Architektura ResNet a porovnání s architekturou VGG-19 [15].

feature maps a vytvoří z nich dvě, které budou mít jako první příznak střední hodnotu resp. maximum z prvních příznaků ze všech feature maps. Zbytek jejich příznaků se doplní analogicky. Nakonec se na ně aplikuje konvoluce a vznikne jedna 2D feature map. Basic unit obsahuje ještě 2 moduly z nichž první spojí opět dohromady všechny feature maps a druhý prohodí jejich pořadí tak, aby se mohli v příštím vstupu do Basic unit rozvětvit jinak.

Vrstva	Velikost výstupu	Počet výstupů
Vstupní obrázek	224×224	3
Konvoluční vrstva + MaxPool	112×112	24
Basic unit 1	28×28	244
Basic unit 2	14×14	488
Basic unit 3	7×7	976
Konvoluční vrstva	7×7	2048
GlobalPool	1×1	
Fully connected		1000

Tabulka 2.2: Architektura ShuffleNetV2.



Obrázek 2.16: Basic unit. (a) Basic unit z ShuffleNetV2. (b) Basic unit z [32] s Mixed attention mechanismem [32].

2.5 Určení věku a aktualizace parametrů

Určení věku probíhá tak, že z poslední vrstvy CNN získáme nějakým postupem jedno číslo, a to prohlásíme za výsledný věk. Během trénování modelu se toto číslo porovná s reálným věkem osoby a aktualizují se parametry CNN. Existující metody určení věku se dají rozdělit do tří směrů:

- Klasifikace
- Regrese
- Ranking methods

2.5.1 Klasifikace

Klasifikace do N věkových skupin spočívá v N neuronech v poslední vrstvě CNN, z nichž se vypočítá výsledný věk. Nevýhoda je ignorování ordinality ve věkových skupinách.

Výpočet věku

Pro určení věku osoby, nebo-li pro výpočet výsledného věku z poslední vrstvy CNN, se používá funkce Softmax. Jedná se o exponenciální funkci, která normalizuje všechny hodnoty x_i z poslední vrstvy neuronů do intervalu 0 až 1 tak, že jejich součet je roven 1. Hodnoty jednotlivých neuronů pak představují pravděpodobnost věku, kterým jsou označkovány. Softmax je definován takto:

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}} \tag{2.7}$$

Aktualizace parametrů

Princip aktualizace parametrů neuronové sítě je následující. Uvažujme velmi jednoduchou neuronovou síť s jednou vstupní vrstvou, jednou výstupní vrstvou, žádnou skrytou vrstvou a binárním klasifikováním (jedná se tedy vlastně o logistickou regresi). Architektura této sítě je na obrázku 2.17. Pro ilustraci uvažujme příklad skladu, který automatizovaně rozesílá jablka a granáty do dvou různých firem. Rozhodnutí jestli je daný předmět jablko modelujeme pomocí podmíněné pravděpodobnosti $P(J|x) = \sigma(xw)$, kde x je vektor příznaků objektu určeného ke klasifikaci, w je vektor vah a σ aktivační funkce, např. logistická sigmoida. Pokud se jedná o granát pak $P(G|x) = 1 - P(J|x)$. Abychom měli co největší úspěšnost klasifikace, snažíme se maximalizovat tuto pravděpodobnost:

$$P(T|X) = \prod_{n=1}^{N} P(t_n|x_n) \tag{2.8}$$

kde T je vektor anotací $t \in \{0, 1\}$ značící jestli jde o jablko nebo granát, X je vektor vektorů příznaků klasifikovaného objektu x a N je počet trénovacích dat. Tento vztah můžeme dále rozepsat takto:

$$\prod_{n=1}^{N} P(t_n|x_n) = \prod_{n=1}^{N} P(J|x_n)^{1-t_n} P(G|x_n)^{t_n} = \prod_{n=1}^{N} (1 - \sigma(x_n w))^{1-t_n} \sigma(x_n w)^{t_n} \tag{2.9}$$

Výraz za druhým rovnítkem v rovnici 2.9 je matematická definice neuronové sítě tohoto skladu. Nyní bychom chtěli, aby tento výraz měl co největší hodnotu, což znamená, že bychom maximalizovali pravděpodobnost správných anotací pro vektor pozorování X.

Cross-Entropy loss je základní objektivní funkce. Místo toho, abychom maximalizovali objektivní funkci, tak minimalizujeme její záporný logaritmus. Pokud využijeme náš příklad skladu, pak výsledná objektivní funkce bude vypadat takto:

$$E(w) = -\sum_{n=1}^{N} (1 - t_n) \ln(1 - \sigma(x_n w)) + t_n \ln(\sigma(x_n w)) \tag{2.10}$$

K nalezení ideálních parametrů w je potřeba funkci z rovnice 2.10 zderivovat (rovnice 2.11) a položit rovnu nule (rovnice 2.12). Tím bychom zjistili, že tato rovnice nemá analytické řešení, takže bychom museli využít nějakou numerickou metodu jako je např. Gradient Descent. Ta spočívá v tom, že nové parametry vypočítáme jako rozdíl starých parametrů a součinu učící konstanty s hodnotou gradientu (rovnice 2.13). Operace rozdíl je zde, protože minimalizujeme, takže chceme jít ve směru proti nejrychlejšímu růstu funkce. Správná

hodnota učící konstanty je klíčová pro konvergenci algoritmu. Běžné neuronové sítě mají více než jeden neuron, tudíž se používá zpětné šíření chyby s více než jedním vektorem vah. Výpočet gradientu chyby v tomto příkladu závisel pouze na jednom vektoru vah w, nicméně reálně používané neuronové sítě mají více vektorů vah a výpočet gradientu chyby je definován rovnicí 2.14.

$$\nabla E(w) = \frac{\delta E}{\delta w} = \sum_{n=1}^{N} (\sigma(x_n w) - t_n) x_n \tag{2.11}$$

$$\nabla E(w) = 0 \tag{2.12}$$

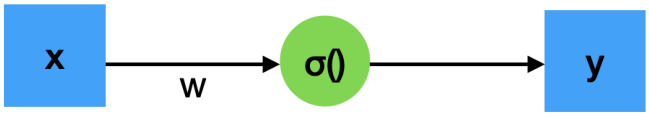
$$w^{\tau+1} = w^{\tau} - \mu \nabla E(w^{\tau}) \tag{2.13}$$

$$\nabla E(w_1, w_2, ..., w_K) = \frac{\delta E}{\delta w_1} \frac{\delta E}{\delta w_2} ... \frac{\delta E}{\delta w_K} \tag{2.14}$$

Obecně pro více tříd je Cross-Entropy loss definována:

$$H(t, p) = -\sum_{i=1}^{N} \sum_{c=1}^{C} t_{i,c} \log p_{i,c} \tag{2.15}$$

kde C je počet klasifikačních tříd, $t_{i,c}$ je 1, pokud dato i reálně patří do třídy c, jinak je 0 a $p_{i,c}$ je pravděpodobnost, že dato i náleží třídě c [7].



Obrázek 2.17: Architektura nejjednodušší neuronové sítě.

Další metoda pro aktualizaci parametrů je Mean-Variance loss [37]. Ta interpretuje věk jako gaussovské rozložení pravděpodobnosti se střední hodnotou a variancí. Princip metody je podobný jako určení věku člověkem, který většinou dokáže interpretovat věk osoby jako nějaké gaussovské rozložení pravděpodobnosti se střední hodnotou a malou variancí. Ty můžeme spočítat takto:

$$m_i = \sum_{c=1}^{C} j p_{i,c} \tag{2.16}$$

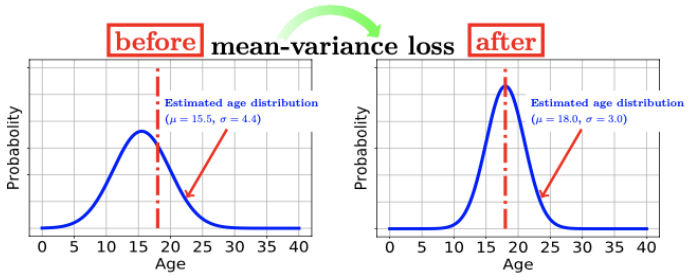
$$v_i = \sum_{c=1}^{C} (c - m_i)^2 p_{i,c} \tag{2.17}$$

kde C je počet tříd a $p_{i,c}$ je pravděpodobnost, že dato s indexem i patří do věkové třídy c. Po aplikování funkce Softmax na poslední vrstvu CNN, se Mean-Variance loss snaží přiblížit střední hodnotu predikovaného rozložení věku k reálnému věku a minimalizovat varianci predikového rozložení, což znamená, že křivka predikovaného věkového rozložení by měla být úzká a špičatá. To je vidět na obrázku 2.18. Predikovaným rozložením věku je myšlena spojitá funkce na jejíž x-ové ose jsou věkové třídy a na y-ové ose jsou hodnoty

neuronů v poslední vrstvě pro jednotlivé věkové třídy. Pro srovnání, Cross-Entropy loss nebere při penalizování v potaz chybné věky, tedy věky, které neodpovídají reálnému věku osoby. Z tohoto plyne, že nedělá rozdíl mezi klasifikováním např. 25leté osoby jako 20leté nebo jako 60leté. Naopak Mean-Variance loss přináší do systému určitou formu ordinality, protože penalizuje i varianci predikovaného rozložení věku. Celková objektivní funkce je definována:

$$\begin{aligned} L_m &= \frac{1}{2N}\sum_{i=1}^{N}(m_i - y_i)^2 \\ L_v &= \frac{1}{N}\sum_{i=1}^{N} v_i \\ L &= L_c + \lambda_1 L_m + \lambda_2 L_v \end{aligned} \tag{2.18}$$

kde N je počet dat, L_c je Cross-Entropy loss a λ_1, λ_2 jsou dva parametry určující vliv daných objektivních funkcí. Systém si je nejjistější v predikovaném věku, o trochu méně si je jistý ve věku ve vzdálenosti $\pm$ 1 od predikovaného věku a míra jistoty ostatních věků na obě strany rychle klesá.



Obrázek 2.18: Ukázka vlivu Mean-Variance loss. Na obrázku vlevo je pravděpodobnostní rozložení věků bez Mean-Variance loss, vpravo pak s použitím Mean-Variance loss [37].

2.5.2 Regrese

Jedná se o jeden neuron v poslední vrstvě CNN, který se učí mapovací funkci $y = f(x)$, kde x je hodnota neuronu a $y \in \mathbb{R}$ je výsledná predikce, zde věk osoby. Regrese reflektuje spojité rozložení věku a ordinalitu, nicméně modeluje věk lineárně, což neodpovídá realitě, protože obličej člověka se nemění lineárně s věkem. Objevují se také metody, které kombinují klafikaci s regresí [32].

Výpočet věku

Při testovaní se jedná o funkci jedné proměnné, tudíž výsledný věk se vypočítá dosazením výstupu CNN do této funkce.

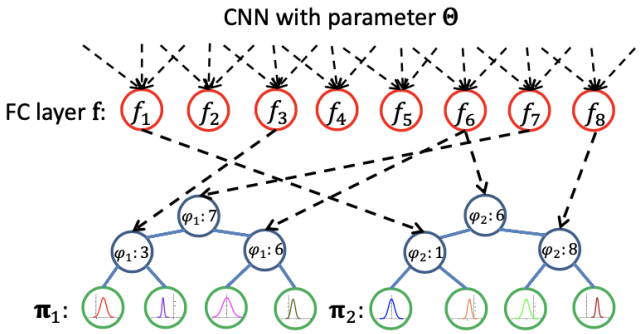
Aktualizace parametrů

Neuronová síť může aproximovat libovolnou vysokodimenzionální nelineární funkci. Oproti klasifikaci zde vystupují jiné objektivní funkce. Často používaná objektivní funkce je Mean Squared Error, která vypadá takto:

$$M(t, y) = \frac{1}{N} \sum_{i=1}^{N} (t_i - y_i)^2 \tag{2.19}$$

kde t_i je anotace data i a y_i je hodnota neuronu data i.

Klíčové je nalézt mapovací funkci mezi hodnotou poslední vrstvy CNN (1 neuronem) a hodnotou reálného věku osoby. V [43] se mapovací funkce modeluje pomocí kolekce rozhodovacích stromů, která tvoří jeden les. Každý strom je tvořen ze Split uzlů (všechny nelistové uzly stromu) a listových uzlů. Před zahájením trénování Index funkce náhodně naváže každý Split uzel na libovolný neuron v poslední fully connected vrstvě CNN. Je tedy možné, že dva Split uzly z různých stromů budou navázané na stejný neuron. Každý strom obsahuje vlastní Index funkci. Split uzly obsahují Split funkci, která na základě hodnoty neuronu na který je uzel navázaný rozhodne, jestli bude trénovací dato posláno do levého, nebo pravého podstromu. Listový uzel obsahuje funkci gaussovského rozložení pravděpodobnosti, která modeluje pravděpodobnosti jednotlivých věků. Určitý integrál všech těchto funkcí v rámci jednoho stromu je roven 1. Každý strom má tedy toto rozložení pravděpodobnosti nezávislé. Výstupem jednoho stromu je věková predikce. Tyto predikce se zprůměrují a získá se výsledná predikce celého lesu. Ilustrace Deep Regression Forest je na obrázku 2.19. Jako loss funkce se zde využívá Negative Log Likelihood loss.



Obrázek 2.19: Příklad Deep Regression Forest. Na obrázku je poslední vrstva neuronové sítě a dva stromy [43].

2.5.3 Ranking methods

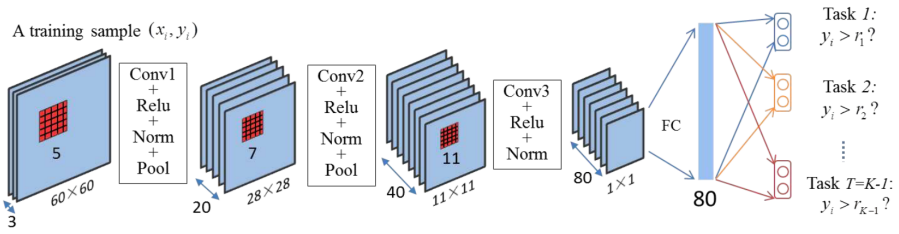
Ranking methods je několik samostatných binárních klasifikátorů (např. CNN s binárními výstupy), které představují věkové skupiny a pomocí jejich výstupů se určí výsledný věk. Tyto metody ovšem trpí velkou výpočetní náročností [12].

[36] modeluje ordinální regresi pomocí série binárních klasifikátorů. Cílem je namapovat vstupní obrázek na nějaký rank. Zde mapují 80 tříd na 79 binárních klasifikátorů (obecně K tříd na $K-1$ binárních klasifikátorů). Každý binární klasifikátor vrací bitovou informaci jestli je osoba na obrázku starší (1), nebo rovna a mladší (0) než věk, který klasifikátor reprezentuje, tzv. rank klasifikátoru. Např. pokud do systému přijde obrázek s osobou ve věku 7 let, pak binární klasifikátory představující věky 1, 2, 3, 4, 5, 6 vrátí 1 a ostatní vrátí 0. Rank q predikovaného věku se vypočítá takto:

$$q = 1 + \sum_{k=1}^{K-1} f_k(x') \tag{2.20}$$

kde $f_k(x') \in \{0, 1\}$ je výstup k-tého binárního klasifikátoru.

Výsledný věk získáme zjištěním, který věk rank q reprezentuje. Obvykle jsou tyto dvě hodnoty totožné. Průběh metody je ukázán na obrázku 2.20.



Obrázek 2.20: Vizualizace metody ordinální regrese [36].

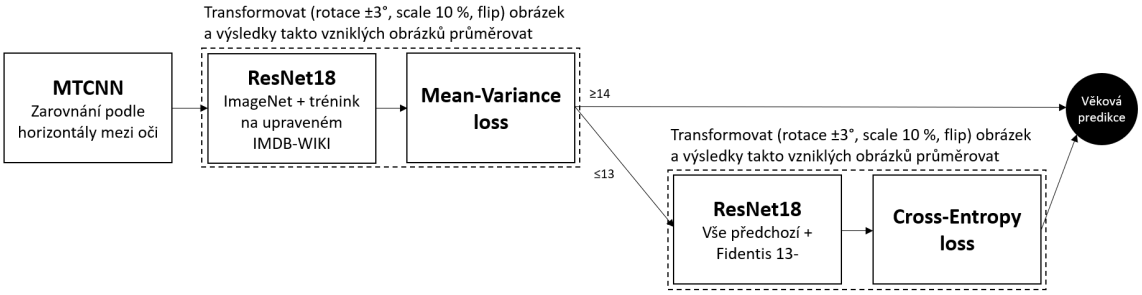
Kapitola 3

Návrh a implementace algoritmu

Tato kapitola vysvětluje návrh algoritmu a také jeho implementaci, včetně vytvoření trénovacího a testovacího datasetu.

3.1 Návrh algoritmu

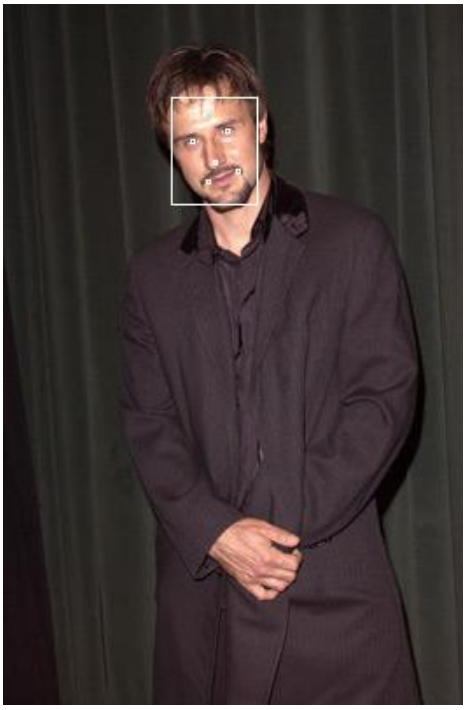
Celistvý návrh algoritmu je ukázán na obrázku 3.1. Níže jsou popsány jeho jednotlivé části.



Obrázek 3.1: Vizualizace návrhu algoritmu. Vlevo je hlavní modul, vpravo je dětský modul.

3.1.1 Předzpracování dat

Jedná se o první fázi algoritmu (obrázek 3.1). Vstupní obrázek bude postoupen obličejovému klasifikátoru, který nalezne obdelník ohraničující obličej a 5 obličejových bodů (obrázek 3.2). Nejprve se výše zmíněný obdelník promění na čtverec, a to tak, že kratší stranu zvětší na obou stranách o odpovídající velikost. Vznikne tedy čtverec, který ohraničuje obličej, nacházející se ve středu obrázku. Ke každé straně se navíc přidá 80 % okolí a ořízne se. Poté je obrázek otočen tak, aby obě oči ležely na jedné horizontální přímce, zmenší se tak, aby obsahoval pouze 40 % okolí a poměrovým přiblížením popř. oddálením vznikne finální snímek obličeje o velikosti 224×224 pixelů (obrázek 3.3). Důvod, proč se přidá 80 % okolí, provede se natočení a zmenší se, aby obsahoval 40 % okolí a ne rovnou oříznutí obrázku s 40% okolím a natočení, je aby se na obrázku nevyskytovala místa s neznámými hodnotami pixelů (rozdíl je vidět na obrázku 3.4). Tyto pixely mohou být nastaveny na jednu barvu nebo mohou opakovat poslední známou barvu, kde ale může nastat, že pokud bude na okraji obrázku vrchol hlavy, tak se touto operací hlava nepřirozeně prodlouží.



Obrázek 3.2: Detekce obdelníkového ohraničení obličeje, levého oka, pravého oka, nosu, levého ústního koutku a pravého ústního koutku [40].

3.1.2 Extrakce a aktualizace parametrů

Získání parametrů ze snímků obličeje zajistí konvoluční neuronová síť. Ta bude předtrénována na datasetu ImageNet. Bude se klasifikovat do 70 věkových tříd, tzn. věky 0-69 let. Algoritmus se skládá ze 2 modulů, kterými jsou hlavní modul a dětský modul. Do obou modulů půjde originální obrázek + jeho 4 transformované varianty, získané aplikováním rotace (2×), přiblížení a flipu na daný obrázek. Predikce modulu se vypočítá jako průměr predikcí těchto pěti obrázků. V prvním zmíněném modulu bude jako objektivní (loss) funkce použita Mean-Variance loss, která dobře reflektuje, že věk je spojitá veličina a zároveň dokáže věky modelovat nelineárně. Tato část bude předtrénována na ImageNet datasetu a natrénována na upraveném IMDB-WIKI datasetu. Pokud v tomto okamžiku bude osoba klasifikována jako 14letá nebo starší, pak je tato predikce konečná. Pakliže tomu bude opačně, obrázek bude ještě předán do dětského modulu, jehož predikce je finální. Dětský modul je odvozen od hlavního modulu s tím, že je dotrénovaný na podmnožině databáze obličejů Fidentis, příslušící věkové skupině 0 až 13 let. Jako loss funkce je použita Cross-Entropy, a to z důvodu, že u dětí jsou věkové změny mnohem výraznější než u dospělých, tudíž jednotlivé věkové třídy jsou si méně podobné. Například 50letý a 55letý člověk jsou si více podobní než 5letý a 10letý člověk, přestože je mezi nimi stejný věkový rozdíl. Všechny predikce se získají z výstupu neuronové sítě, na který se aplikuje Softmax, vynásobí se s příslušnými věky a toto vše se sečte. Ilustrováno je to na obrázku 3.1.

(a) Obrázek před předzpracováním.

(b) Obrázek po předzpracování.

Obrázek 3.3: Ukázka extrakce obličeje z obrázku [40].

3.2 Implementace

V následující části budou stručně popsány vybrané nástroje a knihovny, ve kterých byla implementace realizována. Následovat bude představení implementace trénovacího datasetu a samotného navrhovaného algoritmu a ukázka výsledné aplikace.

3.2.1 Použité nástroje

Zde je stručná charakteristika vybraných použitých knihoven.

PyTorch

PyTorch[1] je volně dostupná open-source knihovna pro strojové učení používaná pro aplikace z oblasti počítačového vidění a zpracování jazyka. Byla vyvinuta společností Facebook's AI Research. Software vytvořený v PyTorch používají firmy jako třeba Uber nebo Tesla. Mezi její výhody patří rychlé výpočty s možností GPU akcelerace, provázanost s jinými používanými Python knihovnami (např. Numpy, Scipy) a nabízí také velmi kvalitní platformu pro tvorbu vlastních neuronových sítí.

Pillow

Python Imaging Library[2] je open-source knihovna umožňující manipulaci s obrázkovými daty. Umožňuje pracovat s jednotlivými pixely, rotovat, vyhlazovat, rozmazávat, upravovat jas či ostrost a mnohé další. Podporuje formáty jako třeba: PPM, PNG, JPEG, GIF, TIFF a BMP.

[1] www.pytorch.org
[2] www.python-pillow.org

(a) Obrázek s 40 % okolí + rotace.

(b) Obrázek s 80 % okolí + rotace + zmenšení na 40 % okolí.

Obrázek 3.4: Ukázka rozdílu rotace při dvou uvedených přístupech [40].

(a) Fotka před transformací.

(b) Fotka po transformaci.

Obrázek 3.5: Ukázka horizontální flip transformace [40].

3.2.2 Vytvoření trénovacího datasetu

Trénovací dataset byl vytvořen z upraveného IMDB-WIKI datasetu a z databáze 3D modelů obličejů Fidentis. Hlavní modul je trénován na upraveném IMDB-WIKI datasetu, dětský modul je trénován navíc na Fidentis 13- datasetu, což je Fidentis dataset shora omezený na věk 13 let.

IMDB-WIKI dataset

Jedná se o největší veřejně dostupný dataset fotografií obličejů osob s věkovou anotací. Databáze vznikla vyhledáním 100000 nejoblíbenějších herců podle webu IMDb[3] a automatickým extrahováním fotografií, data narození, jména a pohlaví z jejich profilu na této stránce. Dále takto procházeli profilové obrázky na Wikipedii[4]. Data u kterých tyto informace nebyly k dispozici odstranili. Podle těchto metadat pak sestavili dataset s věkovými anotacemi. Hodně obrázků jsou statické snímky z filmů, které mohou být staré, a tím pádem mohou trpět nekvalitou nebo černobílým obrazem. Některé obrázky, zejména z IMDb, obsahují více osob, proto ty, kde je druhá nejsilnější detekce obličeje nad prahovou hodnotou,

[3] www.imdb.com

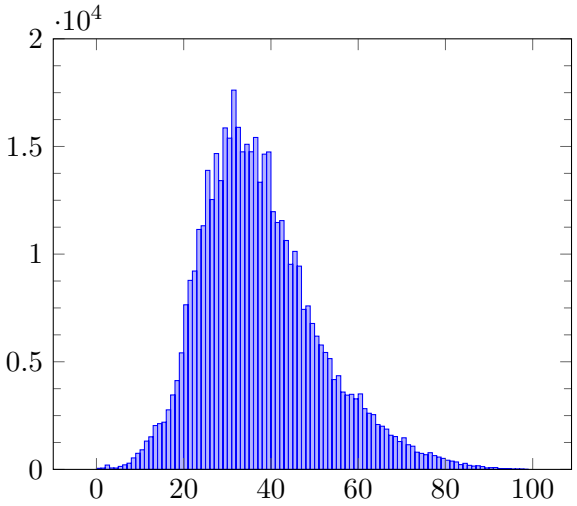
[4] www.wikipedia.org

byly odstraněny. Dataset obsahuje celkem 523051 obrázků obličeje, které tvoří 460723 fotek 20284 osobností z IMDb a 62328 fotek z Wikipedie [40]. Ukázka dat je na obrázku 3.6.

Obrázek 3.6: Ukázka dat z IMDB-WIKI datasetu [40].

Upravený IMDB-WIKI dataset

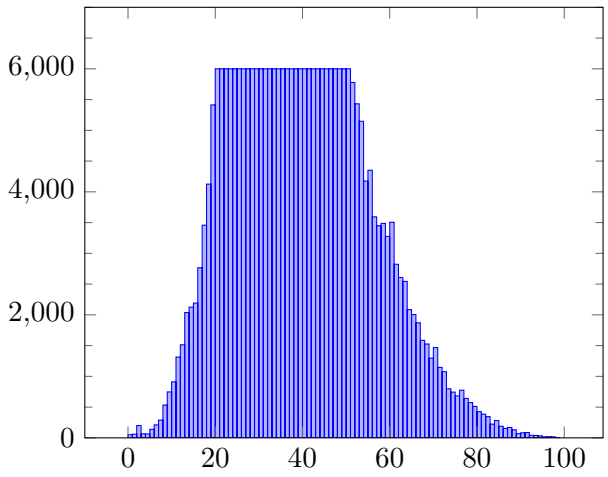
Při procházení IMDB-WIKI datasetu bylo zjištěno, že obsahuje hodně nežádoucích obrázků, např. značku výstrahy, schéma hřiště pro lední hokej, černou tečku o velikosti 1 pixel atp. Dataset obsahuje přes 520000 obrázků, což je velké množství, které představuje velkou časovou a výkonnostní náročnost pro samotné trénování, protože každý obrázek se musí načíst a provést se na něm správné natočení a oříznutí. Tyto dvě skutečnosti byly hlavní motivací pro vznik upraveného IMDB-WIKI datasetu, který tímto zredukoval počet dat na 472 695 o rozměrech 224×224 pixelů. Tyto obličeje jsou již předzpracované dle protokolu ze sekce 3.1.1. Samotné trénování bude nyní rychlejší, protože se během něj nebude provádět předzpracování dat. Rozložení věků je vizualizováno v histogramu 3.7. Implementace vytvoření tohoto datasetu je stejná jako implementace z hlavního programu, která je popsána v sekci 3.2.3.



Obrázek 3.7: Histogram výskytu věků v upraveném IMDB-WIKI datasetu po odstranění neobličejových dat.

Když se podíváme na rozložení dat po věkových skupinách, zjistíme, že kategorie 70 až 100 let je zastoupena velmi sporadicky oproti třeba kategorii 30 až 40 let. Experimentálně jsem ověřil, že pokud se okruh klasifikace sníží ze 100 tříd na 70 tříd, tak se

výsledné predikce zlepší. Je to pravděpodobně zapříčiněno tím, že neuronová sít má z vyšších věkových kategorií k dispozici málo dat, tudíž se je špatně naučí klasifikovat. Další pokus o vylepšení a zpřesnění predikce je, že každý jeden věk bude obsahovat maximálně 6000 snímků, které se vyberou vždy náhodně z celkového počtu obrázků náležících danému věku. Toto sníží nevyrovnanost výskytu jednotlivých věků. Celkově obsahuje zredukovaný dataset 285 870 obličejových dat. Graf výskytu konkrétních věků ve zredukovaném datasetu je vidět v histogramu 3.8. Experimentálně bylo poté ověřeno, že toto zredukování nevede k lepším výsledkům predikce, proto finální verze upraveného IMDB-WIKI obsahuje pouze originální dataset, zbavený fotek bez lidského obličeje o velikosti 472695 dat.



Obrázek 3.8: Histogram výskytu věků ve zredukované verzi upraveného IMDB-WIKI datasetu.

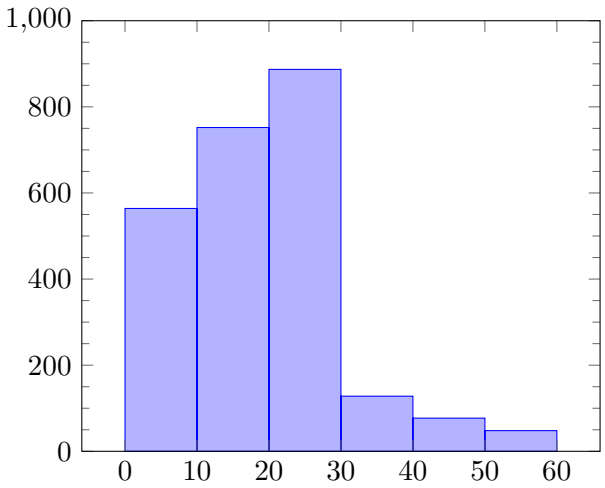
Fidentis dataset

Díky spolupráci s Ústavem antropologie PřF MU mi byla poskytnuta subdatabáze jejich databáze 2476 3D obličejů. Národnostně databáze obsahuje především české a slovenské občany. Jiné národnosti jsou přítomny v řádu jednotek osob. Muži jsou zastoupeni z 47 %, ženy z 53 %. Věkové rozložení je vidět v histogramu 3.9. Z něj je patrné, že databáze obsahuje hodně dětských obličejů, což bývá problém u většiny datasetů. Naopak množství starších dospělých je malé.

Subdatabáze, kterou jsem obdržel čítá 1347 3D modelů, z nichž byla vytvořena 2D data pomocí softwaru SyDa Generátor. Z jednoho modelu se vytvoří 5 obrázků, a to tak, že se model natočí o -10°, -5°, 0°, 5° a 10°. Tato data jsou bohužel neveřejná, tudíž jejich ukázku zde nemohu umístit.

3.2.3 Implementace navrhovaného algoritmu

Navrhovaný algoritmus je implementován v jazyce Python za použití frameworku PyTorch a dalších příbuzných knihoven jako Numpy, Pillow atp. Všechny experimenty, výkonnostní a časové údaje jsou prováděny a jsou platné, pokud nebude uvedeno jinak, na počítači s následující konfigurací: CPU Intel Core i5-9400F (6 jader, 2.9GHz, TB 4.1GHz), GPU Asus TUF-GTX1660-O6G-GAMING (6GB GDDR5), RAM Kingston HyperX Fury Black (16GB DDR4 2666MHz), OS Windows 10 Home.



Obrázek 3.9: Histogram výskytu věkových skupin v databázi Fidentis. Osob starších 60 let se v databázi nachází 14.

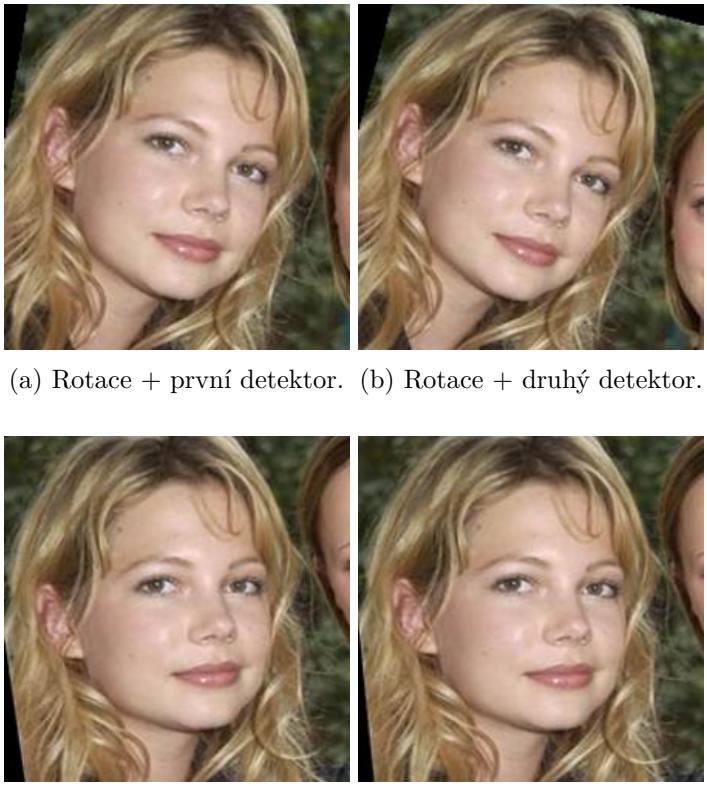
Detektor obličeje

Na detekci obličeje a jeho rysů je použit Multitask Cascaded Convolutional Networks (MTCNN) [49]. Pro jazyk Python jsou volně k dispozici 2 implementace. První[5] je na bázi Tensorflow, druhá[6] pak v PyTorch. Také jsou uvažovány 2 přístupy natočení, buď obrázek natočit o různé úhly (v tomto případě od -45° po 45° s krokem 5°) a vybrat ten s nejvyšším skóre jistoty obličejového klasifikátoru, nebo otočení založené na horizontální přímce mezi očima. Když pak dáme dohromady všechny tyto přístupy vzniknou 4 způsoby implementace předzpracování. Jako metriky pro rozhodnutí byly použity čas a kvalita natočení. Tabulka 3.1 ukazuje časové hodnoty jednotlivých přístupů. Kvalita rotace byla manuálně posuzována na náhodně vybraných obrázcích. Ukázka je na obrázku 3.10. Ve výsledném algoritmu je použit poslední přístup z tabulky 3.1. Rozhodl jsem se tak, protože jeho výsledky jsou velmi přesné a je jednoznačně nejrychlejší, což je při velkém počtu dat důležité. Pokud klasifikátor nenalezne žádný obličej, obrázek bude zahozen. Metody založené na rotaci v tomto případě hrubě selhaly, což může být překvapující. Z literatury plyne, že např. [41] a jiné úspěšné metody tuto techniku využívají, ovšem ne výše zmíněné detektory. Toto může být zapříčiněno tím, že hodnocené obličejové detektory nebyly natrénovány na zarovnaných datech, tudíž zarovnanost dat nemá vliv na jejich výsledné skóre jistoty.

Metoda	Doba předzpracování 1 obrázku [s]
Rotace + první detektor	**15,5**
Rotace + druhý detektor	1,1
Oči na horizontální přímce + první detektor	0,35
Oči na horizontální přímce + druhý detektor	**0,04**

Tabulka 3.1: Časové výsledky metod předzpracování dat.

[5] www.github.com/ipazc/mtcnn
[6] www.github.com/timesler/facenet-pytorch



(a) Rotace + první detektor. (b) Rotace + druhý detektor.

(c) Oči na horizontální přímce + první detektor. (d) Oči na horizontální přímce + druhý detektor.

Obrázek 3.10: Kvalitativní výsledky metod předzpracování dat [40].

Extrakce parametrů

Na extrakci parametrů se v této oblasti v drtivé většině používají konvoluční neuronové sítě. To je také důvod, proč byly zvoleny i pro tento algoritmus. Rozhodl jsem se jít cestou použití již existující CNN, protože existuje mnoho veřejně dostupných sítí s velmi dobrými výsledky a vytvoření vlastní architektury by pokoření již existujících skoro jistě nepřineslo. Pro tento algoritmus byla vybrána architektura ResNet18 s tím, že poslední vrstva byla zmenšena na 70 neuronů. Jedná se o jednu z nejlepších CNN a představuje dobrý kompromis mezi málo hlubokou sítí, kde by hrozilo, že se nenaučí potřebné znalosti ke správnému klasifikování a velmi hlubokou sítí, kde by mohlo nastat přetrénování. PyTorch nabízí její implementaci s možností předtrénování na datasetu ImageNet, což je zde využito. Také byla uvažována síť VGG-16, která má mírně lepší úspěšnost klasifikace, ale má mnohem více parametrů (11 milionů vs. 138 milionů), a tím pádem je pomalejší na trénování i testování.

3.2.4 Aplikace

Tato práce představuje 2 verze algoritmu, tzn. i dvě aplikace, které se liší pouze počtem parametrů argumentu `-m` (2 u první verze, 3 u druhé verze) při testování a množinou povolených parametrů argumentu `-M` (hodnoty 0 nebo 1 u první verze, u druhé verze navíc ještě hodnota 2). Zde bude popsána aplikace pro první verzi algoritmu, nicméně rozdíl je minimální, takže tento popis slouží i pro druhou aplikaci. Jedná se o program s textovým uživatelským rozhraním přijímající následující argumenty:

- `-h, --help` vypíše nápovědu
- `-T [DIR], --train [DIR]` trénovací režim a vybrání adresáře s trénovacími daty
- `-f [FILE], --finetune [FILE]` dotrénovací mód a jméno nového souboru
- `-M INT, --modnum INT` vybere modul, který se bude trénovat
- `-l +INT, --logsfreq +INT` frekvence výpisu tréninkových logů (čím větší hodnota tím méně logů)
- `-E +INT, --epochs +INT` počet trénovacích epoch
- `-r +INT, --randomlopo +INT` trénování pomocí LOPO protokolu se specifikováním kolik iterací se provede
- `-v [DIR], --validate [DIR]` přidání validačního datasetu s uvedením jeho adresáře
- `-t [DIR], --test [DIR]` testovací režim a vybrání adresáře s testovacími daty
- `-e, --evaluate` evaluační mód
- `-j [FILE], --json [FILE]` uloží predikce do souboru ve formátu JSON
- `-L INT, --lopo INT` trénování nebo testování pomocí LOPO protokolu s výběrem vynechané osoby
- `-m FILE, --model FILE` vybere soubor(y) s vahami
- `-p, --preprocessing` předzpracování dat

Nabízí 2 základní režimy provozu: trénování nebo testování.

Trénování

Algoritmus obsahuje 2 moduly, které se dají zvlášť trénovat. Pro trénování musí být aktivní argument `-T` a parametrem argumentu `-M` musí být vybrán modul. Pokud není aktivní argument `p`, tak není aplikováno předzpracování a předpokládá se, že uživatel minimálně zajistí, aby vstupní obrázky měly rozlišení 224×224 pixelů. Aby bylo možno správně extrahovat věk z názvu obrázků, je třeba mít jejich názvy v jednom z povolených formátů, který je např. `A_NB`, kde `A` je řetězec začínající znakem '`f`', `N` je věk a `B` je buď nic, nebo začíná znakem '`_`'. Příklad povoleného názvu je `f_33_osoba1.jpg`. Trénování hlavního modulu daty ze složky `train_dataset` vypadá takto:

```
python3 aenn.py -T train_dataset -M 0 -p
```

Testování

Testování je nutné specifikovat argumentem `-t`. Předzpracování dat a formát názvu souborů funguje stejně jako při trénování. Zde je příklad příkazu pro evaluaci (argument `-e`) systému na datech z adresáře `test_dataset`:

```
python3 aenn.py -t test_dataset -p -e
```

Pokud chceme získat věkovou predikci osob na fotografiích z adresáře `predict_age`, použije se tento příkaz:

```
python3 aenn.py -t predict_age -p
```

Ukázka trénování a testování z aplikace je na obrázku 3.11. Další příklady a podrobnější popis všech funkcí aplikace lze najít v příslušné dokumentaci.

```
Evaluating has started ...

PREPROCESSING
100%|| 18/18 [00:02<00:00,  6.60it/s]

TESTING
100%|| 1/1 [00:00<00:00,  1.58it/s]
testing time: 0m 1s
mae: 3.3889
accuracy: 11%
accuracy +-1: 28%
accuracy +-2: 56%
accuracy +-3: 78%
accuracy +-4: 78%
accuracy +-5: 83%
accuracy +-6: 83%
accuracy +-7: 89%
accuracy +-8: 94%
accuracy +-9: 94%
accuracy +-10: 94%
```

(a) Ukázka evaluace.

```
Testing has started ...

PREPROCESSING
100%|| 50/50 [00:02<00:00, 24.48it/s]

TESTING
100%|| 1/1 [00:00<00:00, 17.29it/s]
testing time: 0m 0s
prediction: {'D:/me.png': 23}
```

(b) Ukázka testování.

```
Training has started ...

LOADING IMAGES
100%|| 983/983 [00:00<00:00, 2322.10it/s]

LOADING IMAGES
100%|| 16/16 [00:00<00:00, 2281.14it/s]

TRAINING
epoch: 1/10, iteration 16/16, loss: 20.7962, acc: 0.0374, mae: 8.1490, val mae: 4.1250, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 2/10, iteration 16/16, loss: 15.4053, acc: 0.0521, mae: 6.8852, val mae: 4.1875, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 3/10, iteration 16/16, loss: 13.3977, acc: 0.0518, mae: 6.3434, val mae: 4.7500, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 4/10, iteration 16/16, loss: 12.4288, acc: 0.0631, mae: 6.0268, val mae: 1.6875, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 5/10, iteration 16/16, loss: 10.7674, acc: 0.0775, mae: 5.3266, val mae: 2.3750, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 6/10, iteration 16/16, loss: 9.8976, acc: 0.0943, mae: 4.8716, val mae: 1.8750, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 7/10, iteration 16/16, loss: 9.3343, acc: 0.0960, mae: 4.6101, val mae: 2.3750, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 8/10, iteration 16/16, loss: 8.7206, acc: 0.1085, mae: 4.5126, val mae: 3.5625, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 9/10, iteration 16/16, loss: 7.7393, acc: 0.1135, mae: 4.0679, val mae: 1.6250, lr: 0.001, inputs: [23, 3, 224, 224]
epoch: 10/10, iteration 16/16, loss: 7.9993, acc: 0.1237, mae: 4.0807, val mae: 0.9375, lr: 0.001, inputs: [23, 3, 224, 224]
training time: 1m 6s
```

(c) Ukázka trénování.

Obrázek 3.11: Prostředí aplikace a její hlavní funkce.

Kapitola 4

Trénování a zhodnocení výsledků

4.1 Trénování

Tato část obsahuje počáteční trénovací experimenty, výběr vhodného místa pro trénování a celkové shrnutí trénování všech částí navrhovaného algoritmu.

4.1.1 Experimenty s trénováním

V této sekci přiblížím počáteční trénovací experimenty na mém PC a z důvodu jeho malé výkonnosti následně uvedu další možnosti trénování.

Počáteční experimenty

Trénování modelu na skoro 500000 datech je výpočetně velmi náročné. Na používaném stroji trvá jednotky dnů. Proto počáteční experimenty, které mají za cíl zjistit vliv předtrénování a barevného či šedotónového vstupu, uvedené v této sekci, budou prováděny na podmnožině upraveného IMDB-WIKI datasetu o počtu 5000 náhodně vybraných obrázků. Validační sada bude taktéž z tohoto datasetu, ovšem o velikosti 500 náhodných dat. Díky těmto experimentům získáme počáteční představu o vlivu jednotlivých faktorů. Společné parametry trénování jsou v tabulce 4.1.

Parametr	Hodnota
Dávka (*Batch size)*	64
Počet epoch (*Epochs)*	30
Chybová funkce (*Loss)*	Mean-Variance
Metoda výpočtu gradientů (*Optimizer)*	Adam

Tabulka 4.1: Společné parametry pro počáteční experimenty.

Nejprve tedy zkusíme trénovat hlavní modul, tzn. model s Mean-Variance loss a bez Fidentis 13- datasetu. Bude porovnán vliv náhodné inicializace neuronové sítě, inicializace vahami z předtrénovaného modelu na ImageNet datasetu a obojí v kombinaci s barevnými či šedotónovými obrázky. Zkusit načítat vstupní obrázky jako šedotónové, je motivováno tím, že na určení věku má hlavní vliv tvar a rysy obličeje nikoliv odstín barvy pleti. Vyhodnocovací metrika experimentů bude MAE (Mean Absolute Error), která je definována takto:

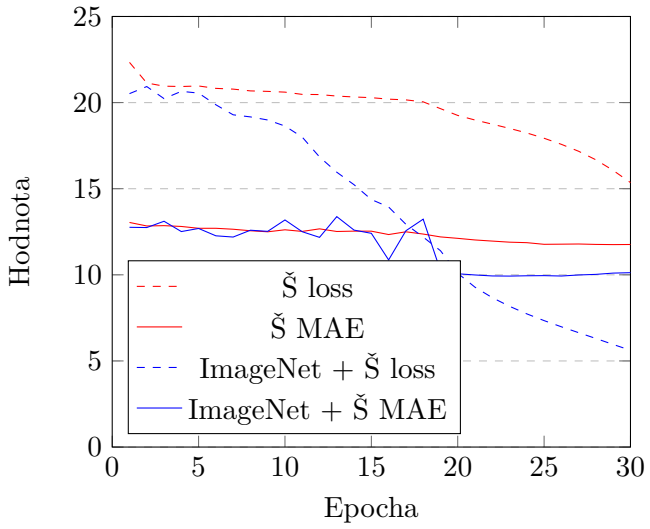
$$MAE = \frac{1}{N} \sum_{i=1}^{N} | \, y_i - t_i \, | \tag{4.1}$$

kde N je počet dat, y_i je predikovaný věk a t_i je reálný věk osoby. Čím menší tato hodnota je, tím lépe. Výsledky experimentů jsou v tabulce 4.2.

Nastavení	Konečná loss	MAE
ImageNet předtrénování + Šedotónový vstup	5,6195	10,1245
ImageNet předtrénování + Barevný vstup	**5,1730**	**10,0265**
Náhodná inicializace + Šedotónový vstup	15,3561	**11,7612**
Náhodná inicializace + Barevný vstup	**17,2045**	11,1531

Tabulka 4.2: Výsledky počátečních trénovacích experimentů.

Z experimentů plyne, že predtrénování na datasetu ImageNet zlepšuje výslednou predikci. Dále lze konstatovat, že barevné obrázky jsou mírně lepší než šedotónové. V grafu 4.1 je vidět průběh validačních MAE a loss funkcí 2 vybraných provedených pokusů. Z výsledků těchto pokusů plyne, že ve finálním modelu budou figurovat barevné obrázky a předtrénování na datasetu ImageNet.



Obrázek 4.1: Průběh 2 vybraných MAE a loss funkcí.

Možnosti trénování

Trénování na mém počítači je prakticky neproveditelné z důvodu jeho častých restartů při náročnějších pokusech. Pokud by trénování proběhlo bez potíží, tak by trvalo větší jednotky dnů. Z těchto důvodů jsem začal hledat alternativy.

Jedna z možností je využít Google Colab[1]. Jedná se o bezplatnou službu od společnosti Google, která uživateli poskytne hardware (včetně GPU) na spuštění Python aplikace. Strojový čas potřebný pro 1 trénovací epochu se snížil asi 1,9×, ale načtení trénovacích

[1] www.colab.research.google.com

obrázků, které probíhá z Google Drive, trvá výrazně delší dobu než při použití lokálního řešení. Další nevýhoda je, že počítač musí být zapnutý během výpočtu a při jakékoliv chybě na straně uživatelského počítače se výpočet zruší.

Jako nejlepší varianta se ukázalo využít MetaCentrum[2]. MetaCentrum představuje Českou Národní Gridovou Infrastrukturu (NGI_CZ), která je součástí Evropské Gridové Infrastruktury (EGI). Účel vzniku byl vytvořit síť propojených výpočetních a úložných strojů akademické sféry. Propojením vznikla možnost počítat vysoce náročné úlohy, které by samostatné pracoviště nezvládlo [1]. Na obrázku 4.2 je ukázána hardwarová infrastruktura MetaCentra.

Obrázek 4.2: Hardwarová infrastruktura MetaCentra [1].

4.1.2 Trénování navrhovaného algoritmu

Níže je charakterizováno trénování všech částí navrhovaného algoritmu a jsou ukázány vybrané pokusy s různými trénovacími parametry.

Trénování hlavního modulu

Hlavní modul byl trénován na upraveném IMDB-WIKI datasetu v MetaCentru. Jako validační dataset byl vybrán FG-NET [38], jelikož je to jeden ze dvou nejpoužívánějších datasetů na validaci algoritmů v této oblasti. Druhý používaný je MORPH-II [39], který nebyl vybrán, jelikož se jedná o placený dataset. FG-NET obsahuje také více dětských dat, tudíž by mohl lépe otestovat přínos dětského modulu navrhovaného algoritmu. Zkoušel jsem různé parametry trénování jako např. plný upravený IMDB-WIKI dataset vs. upravený IMDB-WIKI dataset zredukovaný na maximálně 6000 obrázků z jednoho věku, dětský modul od 0 po 13 let vs. dětský modul od 0 po 17 let, hloubka neuronové sítě atp. Tyto experimenty jsem vyhodnotil, vybrané modely a jejich výkonnost po 10 epochách umístil do tabulky 4.3 a vybral podle nich nejlepší model, jehož statistiky jsou uvedeny v tabulce 4.4.

	FG-NET	FG-NET 25-	FG-NET 26+
ResNet34 + plný upravený IMDB-WIKI	**9,6587**	**8,8131**	13,2356
ResNet18 + zredukovaný upr. IMDB-WIKI	7,9049	**5,922**	**16,2932**
ResNet18 + plný upravený IMDB-WIKI	**7,8188**	6,578	**13,0681**

Tabulka 4.3: Statistiky vybraných variant hlavního modulu.

[2] www.metacentrum.cz

	FG-NET	FG-NET 25-	FG-NET 26+
hlavní modul	6,9069	5,5223	12,7644
s 17- modulem	**7,003**	**5,5297**	**13,2356**
s 13- modulem	6,8168	5,401	12,8063
hlavní modul + transformace	6,7067	5,2785	**12,7487**
s 17- modulem + transformace	6,6466	5,1324	13,0524
s 13- modulem + transformace	**6,4535**	**4,9653**	**12,7487**

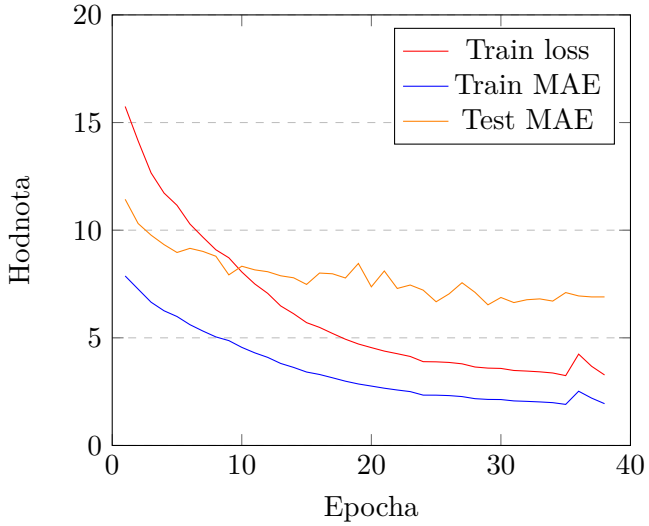
Tabulka 4.4: Statistiky MAE algoritmu a jeho modulů na testovacím datasetu a jeho 2 modifikacích.

Můžeme si všimnout, že nejlepší výsledky má použití transformací a dětského modulu do 13 let včetně. Dále bych podotknul, že toto nejsou výsledky evaluace, pouze otestování algoritmu na datasetu FG-NET. Porovnání zde nemohu nabídnout, protože ostatní algoritmy takovéto testování nedělají, popř. nedělají veřejně. Také si můžeme všimnout, že vyšší věky algoritmus klasifikuje špatně, a to zhruba od 35 let začínají být predikce špatné. Tento fakt bohužel nelze konfrontovat s konkurenčními algoritmy, protože ty nedávají k dispozici úspěšnosti na nějakých věkových podmnožinách datasetů, ale pouze na celých. Tento problém také může způsobovat to, že od jistého věku se obličej mění méně než v dětství, tudíž je težší konkrétní věky rozlišit. Parametry finálního trénovaní jsou uvedeny v tabulce 4.5. Průběh chybové a MAE funkce je pak vidět v grafu 4.3.

Parametr	Hodnota
Dávka (*Batch size)*	64
Počet epoch (*Epochs)*	38
Chybová funkce (*Loss)*	Mean-Variance
Metoda výpočtu gradientů (*Optimizer)*	Adam
Předtrénování (*Pretrained)*	ImageNet

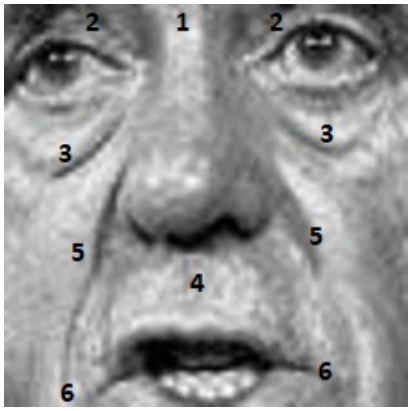
Tabulka 4.5: Parametry pro trénování hlavního modulu.

Pro zlepšení přesnosti klasifikace starších osob mě napadlo vyzkoušet modul, který by klasifikoval na základě vybraných vrásek v obličeji, kterými jsou: horizontální nosní linie, vrásky v oblasti filtra, dolní oční vrásky, horní oční vrásky, nasolabiální rýha a vrásky ústního koutku. Tento modul bude předtrénovaný na hlavním modulu a dotrénovaný na upraveném IMDB-WIKI datasetu, jehož obrázky se ořízno tak, aby obsahovaly pouze oblast, ve které se vyskytují výše zmíněné vrásky. Příklad takového data s vyznačenými vráskami je na obrázku 4.4. Natrénoval jsem 3 modely, které se liší obsaženými daty. První je natrénován všemi daty, druhý 14+ daty a třetí 30+ daty, což se ukázalo jako nejúčinější. Tento modul je navržen pro lepší klasifikaci starších lidí, tudíž bude v navrhovaném algoritmu použit, pokud hlavní modul klasifikuje osobu jako starší nebo rovnu 27 let. Vstupní obrázek do tohoto modulu bude navíc oříznut dle vybraných vrásek (obrázek 4.4). Zkusil jsem různé varianty využití tohoto modulu, ale tato se ukázala jako nejlepší. Vybrané výsledky jsou znázorněny v tabulce 4.6. Je vidět, že přidání tohoto modulu mírně zlepší celkovou úspěšnost (nejlepší výsledek z tabulky 4.4) a v kategorii 26+ zlepší výsledek již znatelně. Vzhledem k tomu, že lidí starších 30 let je v datasetu FG-NET jen velmi málo, tak se lze domnívat, že skutečný přínos by mohl být ještě větší. Nicméně jsou to pouze do-



Obrázek 4.3: Průběh trénování hlavního modulu.

mněnky, proto se budu na tuto variantu navrhovaného algoritmu odkazovat jako na verzi 2 (vizualizace je na obrázku 4.5) a v sekci 4.2 ji porovnám s původní verzí.



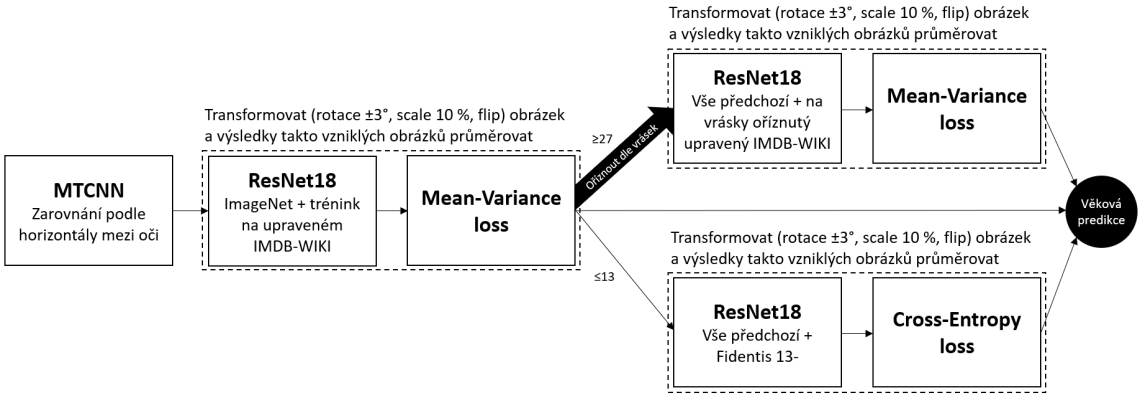
Obrázek 4.4: Ukázka vstupního data 27+ modulu, který klasifikuje věk s důrazem na následující vrásky [40]. 1 – Horizontální nosní linie, 2 – Horní oční vrásky, 3 – Dolní oční vrásky, 4 – Vrásky v oblasti filtra, 5 – Nasolabiální rýha, 6 – Vrásky ústního koutku.

Trénování 13- modulu

Tento modul byl trénován na mém PC s trénovacími parametry uvedenými v tabulce 4.7. Chybová a MAE funkce jsou znázorněny v grafu 4.6. Hranice 13 let pro dětský modul byla vybrána experimentálně. Další varianta byla vytvořit více než 1 specializovaný modul s tím, že specializované moduly by byly vždy dotrénovány na Fidentis subdatabázi, omezené na věkové rozpětí daného modulu. První možnost byla vytvořit 1 hlavní modul (vždy s Mean-Variance loss) a 2 specializované. Hlavní modul by rozhodl, jestli obrázek půjde do 17- modulu s Cross-Entropy loss, nebo do 18+ modulu s Mean-Variance loss a predikce z těchto modulů by již byly konečné. Toto bylo neproveditelné, protože nejvyšší věk ve Fidentis datasetu, který jsem obdržel, je 25 let, tudíž pokusy odhalily, že 18+ modul má

	FG-NET	FG-NET 25-	FG-NET 26+
14+ data	**6,7287**	**4,8911**	**14,5026**
30+ data	6,3293	**5,0879**	11,5812
všechna data	6,6757	4,9233	14,0890
30+ data + vstupní oříznutí dle vrásek	**6,2783**	5,0780	**11,3560**

Tabulka 4.6: Výsledky přidání 27+ modulu do algoritmu a porovnání výsledků s různou množinou trénovacích dat.

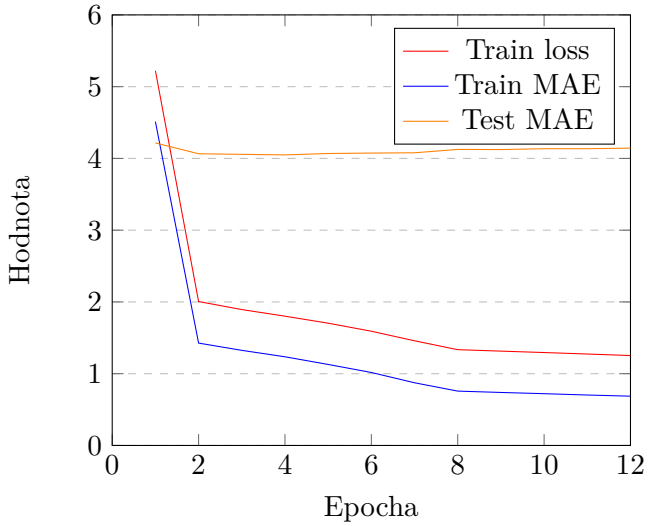


Obrázek 4.5: Vizualizace druhé verze navrhovaného algoritmu.

za této situace negativní vliv. Druhá možnost se skládá z 1 hlavního modulu a 2 specializovaných pro věky 0-13 (použití Cross-Entropy loss) a 14-19 let (použití Mean-Variance loss). S tím, že pokud by hlavní modul stanovil osobě věk vyšší než 19 let, byla by tato predikce finální. Tímto bych více využil potenciál Fidentis datasetu a zároveň netrénoval modul pro všechny dospělé osoby pouze osobami do 25 let. Výsledek tohoto přístupu byl lepší než použití pouze hlavního modulu, ale horší než výsledný návrh algoritmu s jedním hlavním modulem a druhým modulem pro osoby mladší 14 let.

Parametr	Hodnota
Dávka (*Batch size)*	64
Počet epoch (*Epochs)*	12
Chybová funkce (*Loss)*	Cross-Entropy
Metoda výpočtu gradientů (*Optimizer)*	Adam
Předtrénování (*Pretrained)*	Hlavní modul

Tabulka 4.7: Parametry pro trénování 13- modulu.



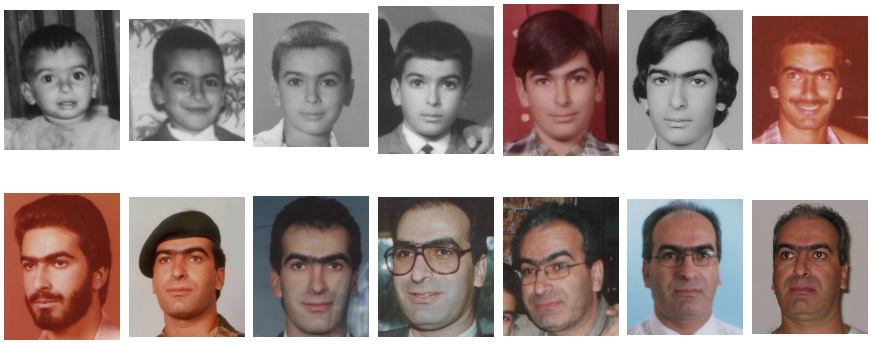
Obrázek 4.6: Průběh trénování dětského modulu.

4.2 Porovnání s aktuálně používanými algoritmy

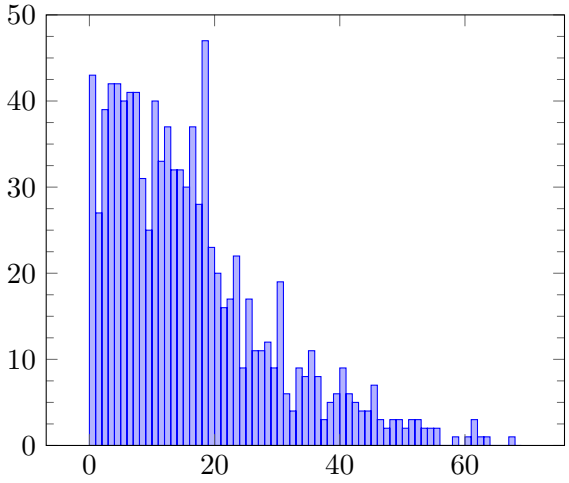
Nejčastěji používané datasety na porovnávání v této oblasti jsou FG-NET a MORPH II.

FG-NET

FG-NET byl vytvořen v roce 2004. Motivace pro jeho vytvoření byla vylepšit stávající metody v oblasti pochopení změn vzhledu obličeje způsobených stárnutím. Databáze se používá i v jiných odvětvích jako je rozpoznávání obličeje nebo mnou vybrané téma určování věku. Databáze je pro akademické účely poskytována zdarma. Obsahuje 1002 obrázků na kterých je 82 různých osob s věkovým rozpětím od 0 po 69 let. Velmi ale převažují mladší osoby zhruba do 25 let. Podíl výskytu starších osob je minoritní. Celkové věkové rozložení datasetu je ukázáno v grafu 4.8. Dataset byl sestaven z velké části naskenováním fyzických fotografií z osobních sbírek daných osob. Některé obrázky mohou být nekvalitní, protože jejich kvalita závisí na dovednostech fotografa, fotoaparátu nebo na použitém fotografickém papíru. Naopak fotky na kterých jsou lidé ve starším věku jsou většinou digitální. Data jsou tedy poměrně variabilní v rozlišení, ostrosti a osvětlení. Na některých snímcích se vyskytují brýle, vousy či klobouky [38]. Ukázka z datasetu je na obrázku 4.7.



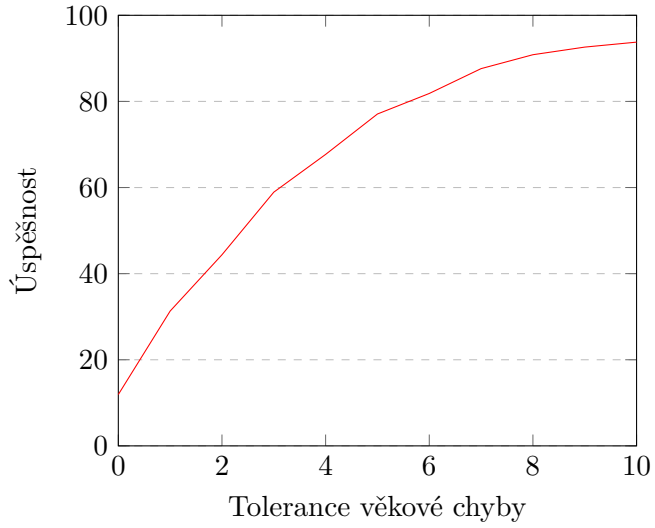
Obrázek 4.7: Ukázka dat jedné osoby z datasetu FG-NET [16].



Obrázek 4.8: Rozložení věků v datasetu FG-NET.

Porovnání

Porovnání algoritmů proběhlo na výše zmíněném datasetu FG-NET, druhý zmíněný dataset je méně vhodný na otestování, protože neobsahuje žádné osoby mladší 14 let, tím pádem důležitá součást algoritmu, dětský modul, by nedostal šanci projevit svůj přínos. Trénování proběhlo dle trénovacího protokolu leave-one-person-out (LOPO). Ten spočívá v tom, že se data rozdělí na data jedné osoby, na kterých se testuje a zbytek, na kterém se trénuje. Hlavní modul byl trénován na všech dostupných datech, dětský modul pak pouze na fotkách s osobou mladší 14 let. V tomto případě máme 82 osob, tudíž se vždy 1 vybere na test a 81 na trénink. Počáteční váhy byly inicializovány z předtrénování na datasetu ImageNet. Trénování každého modelu pro tuto evaluaci proběhlo na 30 epoch a bez augmentace trénovacích dat. Výsledky evaluace jsou v tabulce 4.8 a úspěšnost predikce navrhovaného algoritmu ve variantě, která je v tabulce uvedena jako poslední, je znázorněna v grafu 4.9.



Obrázek 4.9: Průběh úspěšnostní funkce navrhovaného algoritmu na FG-NET datasetu s trénovacím protokolem LOPO.

Algoritmus	MAE	Protokol
AGES [17]	6,77	LOPO
SVR [22]	5,66	LOPO
RED-SVM [9]	5,24	LOPO
DEX [41]	4,63	LOPO
OHRank [10]	4,48	LOPO
MVL [37]	4,1	LOPO
DEX** [41]	3,09	LOPO
MVL** [37]	2,68	LOPO
Navrhovaný algoritmus***	4,5	LOPO*
Navrhovaný algoritmus****	3,88	LOPO*
Navrhovaný algoritmus*****	3,79	LOPO*
Navrhovaný algoritmus	3,82	LOPO*
Člověk [23]	4,7	-

Tabulka 4.8: Porovnání algoritmů na určení věku člověka. *Použito 13 stejných náhodně vybraných osob. **Předtrénování na IMDB-WIKI datasetu. ***Předtrénování na upraveném IMDB-WIKI a Fidentis 13- datasetu. ****Předtrénování pouze poslední fully connected vrstvy na upraveném IMDB-WIKI a Fidentis 13- datasetu. *****Navrhovaný algoritmus verze 2.

Z těchto výsledků plyne jedna zvláštní věc, a to ta, že předtrénování na upraveném IMDB-WIKI zhorší výslednou predikci. Abych se pokusil zjistit proč se tak stalo, vzal jsem si na porovnání 3 varianty hlavního modulu, kterými jsou: váhy inicializované na ImageNet, váhy inicializované na ImageNet + 10 epoch dotrénování na upraveném IMDB-WIKI a váhy inicializované na ImageNet + plné dotrénování (38 epoch) na upraveném IMDB-WIKI. Vybral jsem 4 různé osoby s různou škálou věků a protokolem LOPO natrénoval 12 modelů, jejichž výsledky jsou v tabulce 4.9. Můžeme vidět, že plné předtrénování, tzn. použití nejlepšího natrénovaného modelu validovaného na FG-NET datasetu, vede k horším výsledkům a nižší schopnosti naučit se něco z trénovacího FG-NETu, než využití menšího či žádného předtrénování. Zkusil jsem vypnout trénování všech vrstev kromě poslední fully connected vrstvy a toto vylepšení pomohlo predikci zlepšit.

Také jsem evaluoval druhou verzi navrhovaného algoritmu, a to tak, že 27+ modul jsem natrénoval na FG-NET datasetu, oříznutém podle vybraných rysů obličeje (obrázek 4.4) a zúženém na věky 30 let a vyšší. Trénování tedy proběhlo čistě jen na datasetu FG-NET. Z 13 náhodně testovaných osob mělo pouze 5 fotky s věkem 30 a více let. Zlepšení oproti původnímu algoritmu nastalo, ovšem opět lze spekulovat, že pokud by bylo náhodně vybráno více osob se staršími fotkami, mohlo být zlepšení výraznější. Výsledky všech zde popsaných přístupů jsem také zanesl do tabulky 4.8.

	Osoba 1	Osoba 4	Osoba 6	Osoba 11
0 epoch	**3,6667**	11,3333	**18,9167**	3,4286
10 epoch	5,3333	**10,5833**	**13,0833**	**3,1429**
38 epoch	**7,6667**	**12,5833**	17,9167	**3,7143**

Tabulka 4.9: Porovnání vlivu míry předtrénování na upraveném IMDB-WIKI datasetu na schopnost LOPO trénování na datasetu FG-NET.

Otestování na reálných datech

Pro ukázku reálné funkčnosti jsem vybral 18 kvalitních fotografií osob ze svého osobního archivu, popř. nějaké i z internetu. Jedná se o fotografie mužů i žen, na některých obličejích jsou brýle, jsou foceny v exteriéru i interiéru a během dne či ve tmě. Porovnávám zde (tabulka 4.10) původní algoritmus a verzi 2 navrhovaného algoritmu. Celkové MAE původního algoritmu je 4,9444. MAE u verze 2 navrhovaného algoritmu je pak 2,8889. Z tabulky plyne, že především predikce lidí starších 30 let se velmi zlepšily s přítomností 27+ modulu.

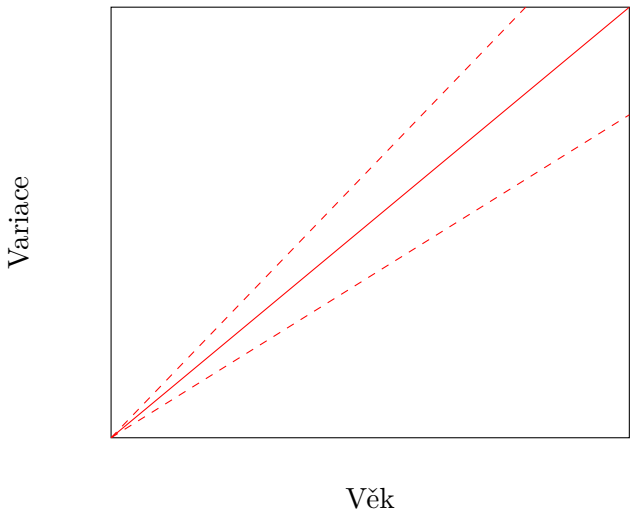
Osoba	Věk	Predikce V1	Predikce V2
Osoba 1	35	**30**	**33**
Osoba 2	37	**32**	**36**
Osoba 3	41	**29**	**35**
Osoba 4	51	**31**	**40**
Osoba 5	11	**11**	**11**
Osoba 6	12	**10**	**10**
Osoba 7	13	**10**	**10**
Osoba 8	17	**20**	**20**
Osoba 9	22	**21**	**21**
Osoba 10	22	**25**	**25**
Osoba 11	23	**22**	**22**
Osoba 12	23	**23**	**23**
Osoba 13	23	**26**	**26**
Osoba 14	23	**25**	**25**
Osoba 15	23	**21**	**21**
Osoba 16	31	**27**	**33**
Osoba 17	39	**28**	**33**
Osoba 18	40	**28**	**36**

Tabulka 4.10: Evaluace původního algoritmu (označeno V1) a verze 2 navrhovaného algoritmu (označeno V2) na datech z mého osobního archivu.

4.3 Možnosti rozšíření

Velké rezervy navrhovaného algoritmu vidím hlavně v klasifikaci starších osob. Nicméně jak jsem již zmínil, konkurenční algoritmy takovéto statistiky nezveřejňují. Tento problém jsem diskutoval s mojí konzultantkou a antropoložkou doc. RNDr. Petrou Urbanovou, Ph.D., která mi přiblížila následující fakta. S přibývajícím věkem dochází postupně ke zvyšování v různorodosti projevů věkově závislých znaků (viz graf 4.10), která snižuje přesnost odhadu věku. Do této skutečnosti se dále přidávají rozdílné podstaty věkových změn. Do 18. až 25. roku života jsou kosti obličeje stále ve vývoji a vztáhnout velikostní a tvarovou změnu v obličeji k věku, je jednodušší než v dospělosti. Naopak dospělý obličej již neroste, mění se pouze měkké tkáně a tyto změny jsou současně více variabilní než ty růstové u dětí.

Jako budoucí směr vidím rozšíření modulu natrénovaném na vybraných rysech obličeje. Takovýto přístup jsem zatím nikde neviděl a dle výsledků testování na databázi FG-NET a mém soukromém archivu, tento postup vypadá zajímavě. Například zkusit jinou techniku než dotrénování původního modelu nebo se zaměřit i na jiné oblasti obličeje.



Obrázek 4.10: Graf různorodosti věkových znaků ve vztahu k věku člověka.

Kapitola 5

Závěr

Cílem této práce bylo prostudovat literaturu týkající se stanovení věku člověka a přidruženou antropologickou literaturu, vytvořit dataset určený k trénování a testování neuronové sítě a navrhnout, implementovat a otestovat expertní systém, který dokáže pomoci v určení stáří člověka z fotografie jeho obličeje.

Teoretická část je rozdělena do tří hlavních částí, kterými jsou předzpracování dat, extrakce parametrů a určení věku a aktualizace parametrů. V každé této části jsou popsány používané metody, kterých podmnožina je součástí touto prací navrhovaného algoritmu. Dále je popsán návrh a implementace algoritmu, během které byly prováděny různé pokusy, které vyústily i ve vznik druhé verze algoritmu.

Evaluace proběhla dvěma způsoby na jednom datasetu FG-NET obsahujícím 1002 obrázků s celkově 82 osobami. První varianta byla, že jsem vždy trénoval na datech 81 osob a testoval na zbylém člověku. Tento způsob zajistí možnost porovnat algoritmus s ostatními, nicméně takto natrénované modely jsou v praxi nepoužitelné, protože jsou natrénovány na velmi malém počtu dat. Nechal jsem vybrat 13 náhodných osob a dosáhl MAE 3,82. U druhé verze navrhovaného algoritmu tato hodnota činila 3,79. Lze se domnívat, že MAE by mohlo být ještě nižší, protože druhá verze navrhovaného algortimu se snaží vylepšit predikce lidí starších 30 let, ale dataset FG-NET takových osob obsahuje málo, tím pádem jeho přínos nemusel dostat šanci se plně projevit. Také, oproti dvěma aktuálně nejlepším algoritmům na určení věku člověka, jsem použil konvoluční neuronovou síť s výrazně menším počtem parametrů, a to z důvodu jejího dobrého poměru mezi schopností extrakce parametrů a rychlostí trénování/testování. Druhý způsob vyhodnocení byl, že jsem navrhovaný algoritmus natrénoval na zhruba 470000 obrázcích a otestoval na celé databázi FG-NET (MAE původní verze 6,4535 a 2. verze pak 6,2783) a navíc také na své soukromé databázi čítající 18 fotografií (MAE původní verze 4,9444 a 2. verze pak 2,8889). Podotknul bych, že ostatní algoritmy čistou evaluaci na FG-NET neprovádějí. Tímto jsem získal představu o možné reálné úspěšnosti predikcí na dvou různorodých datasetech.

Na základě tohoto testování mohu konstatovat, že první verze navrhovaného algoritmu klasifikuje poměrně dobře do zhruba 30 let, poté jsou predikce špatné. Druhá verze navrhovaného algoritmu toto zlepšuje, nicméně u nejstarších jedinců jsou nepřesnosti stále velké. Do budoucna vidím potenciál v rozšíření druhé verze algoritmu a jeho modulu zaměřeného na vybrané obličejové rysy, např. zaměřením se na jinou nebo užší část obličeje. Jako další možnost bych navrhnul zkoumat odlišný způsob kombinace trénování na celoobličejových datech a datech omezených na vybrané obličejové rysy.

Literatura

[1] *MetaCentrum NGI* [online]. 2018 [cit. 3. 3. 2021]. Dostupné z: https://www.metacentrum.cz/cs/index.html/.

[2] *ML Practicum: Image Classification* [online]. 2020 [cit. 22. 1. 2021]. Dostupné z: https://developers.google.com/machine-learning/practica/image-classification/preventing-overfitting?hl=id.

[3] Antipov, G., Baccouche, M., Berrani, S.-A. a Dugelay, J.-L. Apparent Age Estimation From Face Images Combining General and Children-Specialized Deep Learning Models. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2016.

[4] Antipov, G., Baccouche, M., Berrani, S.-A. a Dugelay, J.-L. Effective training of convolutional neural networks for face-based gender and age prediction. *Pattern Recognition*. 2017, sv. 72, s. 15 – 26. DOI: https://doi.org/10.1016/j.patcog.2017.06.031. ISSN 0031-3203. Dostupné z: http://www.sciencedirect.com/science/article/pii/S0031320317302534.

[5] Avula, A. *Medical Image Translation Using Convolutional Neural Networks* [online]. 2020 [cit. 11. 1. 2021]. Dostupné z: https://ysjournal.com/medical-image-translation-using-convolutional-neural-networks/.

[6] Bunkhumpornpat, C., Sinapiromsaran, K. a Lursinsap, C. DBSMOTE: Density-based Synthetic Minority Over-sampling Technique. In: *Applied Intelligence*. 2012, sv. 36, č. 3, s. 664–684. DOI: https://doi.org/10.1007/s10489-011-0287-y.

[7] Burget, L. *Studijní podklady k předmětu SUR – Lineární klasifikátory* [online]. 2020 [cit. 15. 11. 2020]. Dostupné z: https://www.fit.vutbr.cz/study/courses/SUR/public/prednasky/04_lin_klasifikatory/lin_klasifikatory.pdf.

[8] Burget, L. *Studijní podklady k předmětu SUR – Umělé neuronové sítě a Support Vector Machines* [online]. 2020 [cit. 16. 11. 2020]. Dostupné z: https://www.fit.vutbr.cz/study/courses/SUR/public/prednasky/05_neral_networks/NN_CNN_SVM.pptx.

[9] Chang, K.-Y., Chen, C.-S. a Hung, Y.-P. A Ranking Approach for Human Ages Estimation Based on Face Images. In: *2010 20th International Conference on Pattern Recognition*. 2010, s. 3396–3399. DOI: 10.1109/ICPR.2010.829.

[10] Chang, K.-Y., Chen, C.-S. a Hung, Y.-P. Ordinal hyperplanes ranker with cost sensitivities for age estimation. In: *CVPR 2011*. 2011, s. 585–592. DOI: 10.1109/CVPR.2011.5995437.

[11] Chawla, N. V., Bowyer, K. W., Hall, L. O. a Kegelmeyer, W. P. SMOTE: Synthetic Minority Over-sampling Technique. In: *Journal of artificial intelligence research*. 2002, sv. 16, č. 1, s. 321–357. DOI: https://doi.org/10.1613/jair.953.

[12] Chen, S., Zhang, C., Dong, M., Le, J. a Rao, M. Using Ranking-CNN for Age Estimation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, s. 5183–5192.

[13] Deng, J., Dong, W., Socher, R., Li, L., Kai Li et al. ImageNet: A large-scale hierarchical image database. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, s. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[14] Eidinger, E., Enbar, R. a Hassner, T. Age and Gender Estimation of Unfiltered Faces. *IEEE Transactions on Information Forensics and Security*. 2014, sv. 9, č. 12, s. 2170–2179. DOI: 10.1109/TIFS.2014.2359646.

[15] Feng, V. *An Overview of ResNet and its Variants* [online]. 2017 [cit. 6. 3. 2021]. Dostupné z: https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035.

[16] Fu, Y. *FG-NET dataset by Yanwei Fu* [online]. 2014 [cit. 14. 3. 2021]. Dostupné z: https://yanweifu.github.io/FG_NET_data/.

[17] Geng, X., Zhou, Z. a Smith Miles, K. Automatic age estimation based on facial aging patterns. In:. 2007, s. 2234–40.

[18] Gilon, Y. *Convolutional Neural Networks for Visual Recognition* [online]. 2017 [cit. 5. 3. 2021]. Dostupné z: https://cs231n.github.io/convolutional-networks.

[19] Goldmann, T. *SyDa Generator - nástroj pro generování datasetu* [online]. 2021 [cit. 10. 2. 2021]. Dostupné z: http://www.fit.vutbr.cz/~igoldmann/app/sydagenerator/.

[20] Goodfellow, I. *Deep learning*. Cambridge, MA: MIT press, 2016. Adaptive computation and machine learning series. ISBN 978-0-262-03561-3.

[21] Guo, G., Guowang Mu, Fu, Y. a Huang, T. S. Human age estimation using bio-inspired features. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, s. 112–119. DOI: 10.1109/CVPR.2009.5206681.

[22] Guo, G., Fu, Y., Dyer, C. R. a Huang, T. S. Image-Based Human Age Estimation by Manifold Learning and Locally Adjusted Robust Regression. *IEEE Transactions on Image Processing*. 2008, sv. 17, č. 7, s. 1178–1188. DOI: 10.1109/TIP.2008.924280.

[23] Han, H., Otto, C. a Jain, A. Age Estimation from Face Images: Human vs. Machine Performance. In:. červen 2013. DOI: 10.1109/ICB.2013.6613022.

[24] He, K., Zhang, X., Ren, S. a Sun, J. Deep Residual Learning for Image Recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[25] Holibková, A. a Laichman, S. *Přehled anatomie člověka*. 5. vyd. Univerzita Palackého v Olomouci, 2017. ISBN 978-80-244-2615-0.

[26] Jandová, M. *Věkové změny faciální oblasti v ontogenezi člověka*. Brno, CZ, 2010. Bakalářská práce. Přírodovědecká fakulta Masarykovy Univerzity. Dostupné z: https://is.muni.cz/th/avce1/Vekove_zmeny_facialni_oblasti_v_ontogenezi_cloveka.pdf.

[27] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J. et al. Caffe: Convolutional Architecture for Fast Feature Embedding. In: *Proceedings of the 22nd ACM International Conference on Multimedia*. New York, NY, USA: Association for Computing Machinery, 2014, s. 675–678. MM '14. DOI: 10.1145/2647868.2654889. ISBN 9781450330633. Dostupné z: https://doi.org/10.1145/2647868.2654889.

[28] Kalášková, L. *Hodnocení nemetrických znaků lidského obličeje na nestandardním 2D a 3D záznamu*. Brno, CZ, 2018. Diplomová práce. Přírodovědecká fakulta Masarykovy Univerzity. Dostupné z: https://is.muni.cz/th/io7dx/Diplomova_prace.pdf.

[29] Krizhevsky, A., Sutskever, I. a Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In: Pereira, F., Burges, C. J. C., Bottou, L. a Weinberger, K. Q., ed. *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2012, sv. 25. Dostupné z: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[30] Lapuschkin, S., Binder, A., Muller, K.-R. a Samek, W. Understanding and Comparing Deep Neural Networks for Age and Gender Classification. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*. 2017.

[31] Lecun, Y., Bottou, L., Bengio, Y. a Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*. 1998, sv. 86, č. 11, s. 2278–2324. DOI: 10.1109/5.726791.

[32] Liu, X., Zou, Y., Kuang, H. a Ma, X. Face Image Age Estimation Based on Data Augmentation and Lightweight Convolutional Neural Network. *Symmetry*. MDPI AG. 2020, sv. 12, č. 1, s. 146. DOI: 10.3390/sym12010146. ISSN 2073-8994. Dostupné z: http://dx.doi.org/10.3390/sym12010146.

[33] Ma, N., Zhang, X., Zheng, H.-T. a Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[34] Mathias, M., Benenson, R., Pedersoli, M. a Van Gool, L. Face Detection without Bells and Whistles. In: Fleet, D., Pajdla, T., Schiele, B. a Tuytelaars, T., ed. *Computer Vision – ECCV 2014*. Cham: Springer International Publishing, 2014, s. 720–735. ISBN 978-3-319-10593-2.

[35] Nayak, S. *Understanding AlexNet* [online]. 2018 [cit. 2. 3. 2021]. Dostupné z: https://learnopencv.com/understanding-alexnet/.

[36] Niu, Z., Zhou, M., Wang, L., Gao, X. a Hua, G. Ordinal Regression With Multiple Output CNN for Age Estimation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[37] Pan, H., Han, H., Shan, S. a Chen, X. Mean-Variance Loss for Deep Age Estimation From a Face. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[38] Panis, G., Lanitis, A., Tsapatsoulis, N. a Cootes, T. F. Overview of research on facial ageing using the FG-NET ageing database. *IET Biometrics*. Institution of Engineering and Technology. 2016, sv. 5, s. 37–46(9). ISSN 2047-4938. Dostupné z: https://digital-library.theiet.org/content/journals/10.1049/iet-bmt.2014.0053.

[39] Ricanek, K. a Tesafaye, T. MORPH: a longitudinal image database of normal adult age-progression. In: *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. 2006, s. 341–345. DOI: 10.1109/FGR.2006.78.

[40] Rothe, R., Timofte, R. a Gool, L. V. *IMDB-WIKI – 500k+ face images with age and gender labels* [online]. 2015 [cit. 5. 10. 2020]. Dostupné z: https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/.

[41] Rothe, R., Timofte, R. a Gool, L. V. Deep Expectation of Real and Apparent Age from a Single Image Without Facial Landmarks. *International Journal of Computer Vision*. 2018, s. 144–157.

[42] Schmeling, A., Dettmeyer, R., Rudolf, E., Vieth, V. a Geserick, G. Forensic age estimation—-methods, certainty, and the law. *Deutsches Arzteblatt international*. 2016, sv. 113, s. 44–50. ISSN 1866-0452.

[43] Shen, W., Guo, Y., Wang, Y., Zhao, K., Wang, B. et al. Deep Regression Forests for Age Estimation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[44] Simonyan, K. a Zisserman, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.

[45] Sun, X., Wu, P. a Hoi, S. C. Face detection using deep learning: An improved faster RCNN approach. *Neurocomputing*. 2018, sv. 299, s. 42 – 50. DOI: https://doi.org/10.1016/j.neucom.2018.03.030. ISSN 0925-2312. Dostupné z: http://www.sciencedirect.com/science/article/pii/S0925231218303229.

[46] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. et al. Going Deeper With Convolutions. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[47] Urbanová, P., Ferková, Z., Jandová, M., Jurda, M., Černý, D. et al. Introducing the FIDENTIS 3D Face Database. *Anthropological Review*. Berlin: Sciendo. 01 Jun. 2018, sv. 81, č. 2, s. 202 – 223. DOI: https://doi.org/10.2478/anre-2018-0016. Dostupné z: https://content.sciendo.com/view/journals/anre/81/2/article-p202.xml.

[48] Wong, S. C., Gatt, A., Stamatescu, V. a McDonnell, M. D. Understanding Data Augmentation for Classification: When to Warp? In: *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. 2016, s. 1–6. DOI: 10.1109/DICTA.2016.7797091.

[49] Zhang, K., Zhang, Z., Li, Z. a Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*. 2016, sv. 23, č. 10, s. 1499–1503. DOI: 10.1109/LSP.2016.2603342.

Příloha A

Obsah přiloženého paměťového média

Struktura souborů na paměťovém médiu je následující:

```
/
├── v1/
│   ├── aenn.py
│   ├── aennlib.py
│   ├── weights0.pth
│   └── weights1.pth
├── v2/
│   ├── aenn.py
│   ├── aennlib.py
│   ├── weights0.pth
│   ├── weights1.pth
│   └── weights2.pth
├── doc-v1/
│   ├── doc-v1.pdf
│   └── *
├── doc-v2/
│   ├── doc-v2.pdf
│   └── *
├── output/
│   └── predictions.json
├── fg-net/
│   └── *
├── fg-net-m0/
│   └── *
├── fg-net-m1/
│   └── *
├── thesis/
│   ├── BP.pdf
│   └── *
└── readme.md
```