

**Jihočeská univerzita
v Českých Budějovicích**

Přírodovědecká fakulta



Identifikace osob pomocí analýzy chůze

Bakalářská práce

Filip Vejšický

Vedoucí práce: Ing. Jiří Jelínek, CSc.

České Budějovice 2022

Bibliografické údaje

Vejšický, Filip, 2022: Identifikace osob pomocí analýzy chůze. [Identification of a person using gait analysis, Bc. Thesis, in Czech] – 67 p., Faculty of Science, The University of South Bohemia, České Budějovice, Czech Republic.

Annotation

This thesis deals with the use of gait for person identification. The main objective of the thesis is to design and implement a system that will perform real-time automatic identification of humans using an RGB camera. The thesis also includes a survey of state-of-art methods and approaches to gait recognition. The system will offer a complete solution using state-of-art technology, that will not need any special equipment. Emphasis is placed on low hardware requirements, simple use, and the possibility of practical usage.

Key words

gait biometrics, gait recognition, biometric system, automatic gait recognition, framework, gait, Python, Keras, Tensorflow, deep learning, machine learning

Prohlašuji, že svoji diplomovou/bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

V Českých Budějovicích dne 8. 12. 2022

Filip Vejšický

Vejšický

Poděkování

Rád bych poděkoval vedoucímu práce Ing. Jiřímu Jelínkovi, CSc. za vstřícnost, čas věnovaný konzultacím a užitečné rady. Dále chci poděkovat všem blízkým za podporu.

Obsah

1 Úvod	1
1.1 Cíle práce	2
2 Teoretická část	3
2.1 Biometrika	3
2.1.1 Rozdělení.....	3
2.1.2 Použití	4
2.1.3 Požadavky	5
2.1.4 Přesnost biometrického systému	6
2.1.5 Výhody a nevýhody	7
2.1.6 Zákon.....	7
2.2 Biometrika chůze	8
2.2.1 Snímače	9
2.2.2 Vlastnosti.....	9
2.2.3 Využití.....	10
2.3 Způsoby reprezentace chůze	11
2.3.1 Model based přístupy	11
2.3.2 Model free přístupy	12
2.4 Tradiční metody	13
2.4.1 K-Nearest Neighbours (KNN)	14
2.4.2 Support Vector Machine (SVM).....	14
2.6 Hluboké učení	15
2.6.1 Konvoluční neuronové sítě.....	16
2.6.2 Siamské neuronové sítě.....	18

2.6.3 Rekurentní neuronové sítě.....	20
2.6.4 Autoenkodéry	21
2.6.5 Generativní soupeřící sítě.....	22
2.6.6 Kapslové sítě	22
2.5 Podobné existující aplikace	23
3 Praktická část.....	24
3.1 Popis řešení.....	24
3.2 Struktura systému.....	27
3.3 Použité technologie	27
3.4 Datasets.....	29
3.5 Části systému.....	30
3.5.1 Detekce objektů.....	31
3.5.2 Sledování objektů.....	33
3.5.3 Vytváření siluet	36
3.5.4 Detekce cyklu chůze.....	41
3.5.5 Detekce směru	43
3.5.6 Rozpoznání osob	45
3.6 Testování.....	49
4 Závěr	52
4.1 Zhodnocení výsledků	52
4.2 Rozšíření a vylepšení	52
Seznam literatury	54
Seznam obrázků.....	65
Seznam tabulek.....	66
Přílohy	67

Seznam zkratek

FAR – False Acceptance Ratio

FRR – False Rejection Ratio

FP – False positive

TP – True positive

FN – False negative

TN – True negative

TAR – True Acceptance Rate

EER – Equal Error Rate

ROC – Receiver Operation Characteristic křivka

GEI – Gait Energy Image

KNN – K -Nearest Neighbour

NN – Nearest Neighbour

SVM – Support Vector Machine

CNN – konvoluční neuronová síť

RNN – rekurentní neuronová síť

LSTM – Long Short-Term Memory síť

GRU – Gated Recurrent Unit síť

GAN – Generativní soupeřící síť

ORM – Objektově relační mapování

IoU – Intersection over Union

IoT – Internet of Things

NMS – Non Max Suppression

1 Úvod

V oblasti využití chůze jako biometrické charakteristiky pro rozpoznávání osob probíhá neustálý výzkum, který si klade za cíl zdokonalení a rozšíření této technologie s unikátními a velmi zajímavými vlastnostmi, které z ní dělají atraktivní problematiku s velkým potenciálem. Velký posun ve výzkumu začal v roce 2014 s rozšířením hlubokých neuronových sítí. Ty velkou měrou zvýšily dynamičnost vývoje v oblasti a v této práci je jim ze všech metod věnovaný největší prostor. V současné chvíli je stále identifikace pomocí chůze v praxi využívána velmi zřídka, a to především ve srovnání s ostatními biometrickými metodami. Přínos této biometrické metody by se projevil hlavně v oblasti bezpečnosti a forenzní analýzy, ovšem neomezil by se jen na ně. Z těchto důvodů je téma v současnosti aktuální a velmi aktivní.

Tato bakalářská práce si dává za cíl přispět k rozšíření identifikace pomocí analýzy chůze vypracováním rešerše současných metod, a především vytvořením volně dostupného systému, který bude použitelný jako framework nebo knihovna pro Python.

Systém nabídne kompletní implementaci automatické identifikace osob pomocí chůze v reálném čase z videozáznamu běžné kamery. Cílem je vytvoření systému pro základní použití a poskytnutí jednoduchého modulárního nástroje pro vývojáře. Systémem poskytované řešení bude charakterizované nízkou náročností na zařízení a využitím aktuálních přístupů a metod. Důraz je kladen na možnost využití v různých podmínkách reálného světa, a především na dostupnost pro široký okruh uživatelů. Vzhledem k tomuto zaměření je předpokládáno využití hlavně v nekritických situacích, kde není vyžadována maximální spolehlivost a chybný úsudek nevyústí v zásadní škody. Takové využití může být nalezeno například v domácnosti nebo u docházkového systému. Systém samozřejmě může sloužit jako způsob seznámení s technologií nebo jako základ a inspirace pro další systémy. K tomu přispívá modulárnost systému a možnost využití jeho jednotlivých částí nebo naopak snadné nahrazení libovolných modulů svými vlastními.

Tato práce reaguje na nedostatek nástrojů se stejným zaměřením. Volně dostupných systémů, které by byly srovnatelné existuje velmi malé množství. Tyto existující systémy se od navrženého liší používanými metodami, rozsahem i zaměřením. Obvykle se jedná spíše o ukázky nebo experimenty než nástroje určené k praktickému využití. V důsledku těchto skutečností mají odlišné vlastnosti.

Rešerše obsažená v této práci se snaží přiblížit čtenáři oblast identifikace pomocí chůze a usnadnit mu orientaci v současných metodách. V úvodu obsahuje také související informace z oblasti biometriky, které zasazují téma do širšího kontextu.

1.1 Cíle práce

Konkrétní cíle a výstupy práce, které již byly naznačeny v úvodu jsou shrnuty následujícími body.

- Rešerše aktuálních metod používaných pro identifikaci osob pomocí analýzy chůze, a to především metod založených na hlubokých umělých neuronových sítích.
- Návrh řešení automatické identifikaci osob pomocí analýzy chůze v reálném čase z videozáznamu běžné kamery a analýza použitých metod.
- Návrh systému, který bude obsahovat navržené řešení, v souladu s principy návrhu software včetně použití UML diagramů.
- Implementace navrženého systému v programovacím jazyce Python.
- Testování a ověření funkčnosti řešení.
- Rozbor získaných poznatků a popsání návrhů na úpravu a zdokonalení.

Prvním bodem se zabývá teoretická část práce. Další body s výjimkou posledního tvoří obsah praktické části. Poslední bod může čtenář nalézt v závěrečné části.

2 Teoretická část

2.1 Biometrika

Biometrika (také nazývaná biometrie) je věda zabývající se rozlišováním jednotlivců pomocí jejich vlastností. Technologický vývoj umožnil provádět biometriku pomocí automatických systémů. Vývoj byl motivovaný potenciálem technologie pro autentizaci nabízející atraktivní výhody oproti tradičním metodám jako například heslo nebo čipová karta, především v oblasti omezení přístupu ke zdrojům. Hlavní výhody biometrických metod vychází z podstaty toho, že se jedná o součást každé osoby a není tedy nutné žádné přidané úsilí (jako například zapamatování si hesla), napodobení nebo odcizení u biometrických znaků není možné nebo je výrazně obtížnější a méně obvyklé. Tyto výhody mají velký význam pro uživatelské pohodlí a bezpečnost. Přínos se ovšem nezastavil u autentizace a našel využití v dalších technologiích souvisejících s identitou. Význam biometrických systémů stále roste spolu s množstvím služeb a zařízení vázaných na digitální identitu uživatele a novými možnostmi jejich využití. S významem digitální identity a rozšířením biometrických systémů ovšem roste nebezpečí zneužití biometrických údajů.

Aplikace biometrie lze zařadit do dvou kategorií – identifikace a verifikace. Při identifikaci dochází k přiřazení identity neznámému objektu hledáním shody porovnáváním biometrických vlastností se všemi uloženými identitami. Verifikace slouží k ověření identity nárokové objektem. Biometrika umožňuje k tomu využít vlastnosti jedince na rozdíl od tradičních metod založených na prokázání znalosti nebo vlastnictví.

2.1.1 Rozdělení

Biometrické vlastnosti lze rozdělit na fyzické a behaviorální. Mezi fyzické se řadí vlastnosti fyzických charakteristik našeho těla. Patří mezi ně například otisk prstu, duhovka, sítnice, vzhled obličeje a krevní řečiště. Behaviorální vlastnosti jsou takové, které zachycují vzorce chování. Do nich spadá dynamika podpisu, interakce s určitými systémy, hlas a chůze. Některé vlastnosti mohou patřit do obou kategorií, protože kombinují chování s fyzickými vlastnostmi lidského těla. Takovými vlastnostmi jsou hlas a chůze.

Další rozdělení biometrických znaků odděluje tzv. měkké biometrické znaky od klasických. Jako měkké biometrické znaky jsou označovány ty charakteristiky, které udávají informaci o jednotlivci, ale nejsou samostatně dostatečné pro jednoznačné určení identity. Tato kategorie vychází z přirozených způsobů, kterými se lidé identifikují. Často používané vlastnosti z této kategorie jsou výška, váha, pohlaví, věk a barva vlasů. Měkké biometrické znaky mohou ve správné kombinaci plnit roli jednoznačně identifikovat jednotlivce anebo zdokonalit přesnost klasické biometrie. [2]

System využívající více biometrických vlastností se nazývá multimodální. Využití více vlastností v multimodálních systémech může mít několik důvodů. Například zvýšení přesnosti, spolehlivosti a vylepšení nedostatků technologie nebo vlastností. V důsledku toho zvýší bezpečnost systému nebo umožní použití ve scénářích, které unimodální systém neumožňuje.

2.1.2 Použití

Biometrické systémy nachází své uplatnění v širokém spektru oblastí. Nejběžněji jsou používány jako metoda autentizace za účelem zabezpečení aktiv. V současnosti je možné se s biometrickými systémy v této roli setkat u osobních zařízení jako jsou mobilní telefony a osobní počítače. Dále u internetových služeb, aplikací a smart bankingu. Mimo digitální aktiva může být obdobným způsobem omezený vstup do budovy nebo jiných míst, kde je vyžadované omezení přístupu.

Autentizace není jediným způsobem použití v kontextu bezpečnosti. Biometrické vlastnosti se stávají součástí různých dokladů jako jsou občanské průkazy, pasy a další vládní dokumenty. Mohou proto sloužit k ověření identity vlastníka a omezovat tím krádeže identity nebo padělaní těchto dokumentů. Biometrika použitá jako součást autorizace pomáhá bránit proti podvodům. V tomto kontextu lze využít chování uživatele při interakci se službou a zvýšit bezpečnost detekováním podezřelého chování.

Mimo oblast bezpečnosti nachází biometrika přirozené využití v oblasti forenzních věd a kriminalistiky. V současnosti je použití biometrických systémů v soudních procesech

omezeno. Jejich výstupy v mnoha případech nebývají přípustné jako důkazy, ale mohou sloužit jako součást jiného důkazu. Kriminalistům tyto systémy pomáhají při vyšetřování. Především zjišťováním a ověřováním identit podezřelých osob. [1]

Vzhledem k jednoznačnosti a obtížnému obcházení je biometrika vhodná pro použití v docházkových systémech.

Některé biometrické systémy lze aplikovat v personalizaci. Při interakci se systémem nebo službou, kde je určitá biometrická vlastnost zaznamenána neinvazivním způsobem, umožňuje pohodlným způsobem tuto interakci personalizovat a zvýšit tím uživatelský komfort. Rozšířené je toto použití u hlasových asistentů v domácnosti jako je Google Assistant a Amazon Echo, u hlasového ovládání v autech a dalších zařízeních jako jsou telefony a osobní počítače. Na základě hlasu identifikují uživatele a na příkazy reagují přizpůsobenou formou.

2.1.3 Požadavky

Při výběru biometrické vlastnosti je nutné zohlednit požadavky, které budou na systém kladeny. A. K. Jain et al. [3] kategorizovali tyto požadavky do těchto skupin, které určují vhodnost každé biometrické vlastnosti.

- Univerzálnost – Vlastnost má každá osoba v populaci s přijatelným množstvím výjimek.
- Unikátnost – Rozdíly napříč populací jsou dostatečné a případů, kdy není rozlišení jedinců možné, je přijatelné množství.
- Neměnnost – V průběhu času nedochází k výrazným změnám nebo k nim dochází přijatelně často.
- Měřitelnost – Musí být možné zachytit a digitalizovat vlastnost pomocí vhodného zařízení splňující veškeré požadavky způsobu užití. Mezi tyto požadavky mohou patřit například uživatelská přívětivost, schopnost pracovat v různých podmínkách, rychlost zpracování nebo přesnost.
- Přijatelost – Uživatel systému musí být ochotný biometrickou vlastnost systému poskytnout.

- Napodobitelnost – Vlastnost by nemělo být možné napodobit.

Většina biometrických znaků nedokáže dokonale splnit veškeré požadavky, proto je výběr vhodného biometrického systému závislý na způsobu použití a z něj vyplývajících priorit.

2.1.4 Přesnost biometrického systému

Na rozdíl od autentizace pomocí hesla dochází u biometrických systémů k přesné shodě výjimečně. Data zachycená snímačem nemusí být naprosto totožná, odlišné podmínky mohou ovlivnit proces sejmutí dat a samotný snímač také nezachytí data vždy naprosto totožně. Z toho důvodu obvykle ani data reprezentující jednu totožnou vlastnost od stejné osoby nebývají naprosto shodná. Data zachycující biometrickou vlastnost v digitální podobě se nazývají biometrická šablona.

Míru shodnosti dvou biometrických šablon vyjadřuje skóre podobnosti. To může být vyjádřeno různými metrikami (například Euklidovskou vzdáleností). Pro určení shody je nutné zvolit práh skóre podobnosti, který rozhodne, kdy jsou šablony považovány za shodné. Procento z případů šablon odlišných osob, kdy se jejich skóre podobnosti šablon nachází pod prahem se nazývá False Acceptance Rate (FAR). Podobně procento z šablon totožné osoby, kdy skóre podobnosti naopak převyšuje práh se označuje False Rejection Rate (FRR). Změnou prahu měníme FAR i FRR. Z principu není možné snížit obě hodnoty zároveň.

Další související pojmy:

- False positive (FP) – odlišná osoba je vyhodnocena chybně jako shodná
- True positive (TP) – shodná osoba je vyhodnocena správně jako shodná
- False negative (FN) – shodná osoba je vyhodnocena jako odlišná
- True negative (TN) – odlišná osoba je vyhodnocena jako odlišná
- True Acceptance Rate (TAR) – procento z šablon shodných osob vyhodnoceno správně, negace FAR

Vztah FAR a FRR hodnot systému může být dobře znázorněn pomocí Receiver Operation Characteristic (ROC) křivky. ROC křivka zobrazuje FAR a TAR nebo FRR (sémanticky

totožná převrácená křivka) v různých hodnotách prahu. Bod, kde se FAR a TAR (FRR) rovnají se nazývá EER. EER vyjadřuje výkon systém a využívá se k porovnávání systémů. V případě použití FAR a TAR znamená vyšší EER lepší výsledek. Další metrikou výkonu systému je obsah oblasti pod křivkou označovaný jako Area Under The Curve (AUC). Obě metriky jsou nezávislé na zvoleném prahu, a právě to je činní vhodnými pro porovnávání výkonu systémů.

2.1.5 Výhody a nevýhody

Shrnutí typických obecných výhod a nevýhod použití biometrický znaků pro určení identity.

Výhody:

- Nelze jednoduše odcizit
- Nelze zapomenout/ztratit
- Nelze popřít
- Obtížné napodobit

Nevýhody:

- Nelze jednoduše změnit
- False Positive a False Negative chyby
- Vyžaduje zařízení schopné zachytit biometrický znak
- Biometrický znak je v každém systému stejný
- Napodobení může mít větší následky

2.1.6 Zákon

Biometrické údaje spadají v rámci Obecného nařízení o ochraně osobních údajů (GDPR) Evropské unie do zvláštní kategorie osobních údajů a jejich zpracování na území Evropské unie je na výjimky zakázané [4].

Podle článku 9 GDPR je zpracování osobních údajů, které vypovídají o rasovém či etnickém původu, politických názorech, náboženském vyznání či filozofickém přesvědčení nebo členství v odborech, a zpracování genetických údajů, biometrických údajů za účelem jedinečné identifikace fyzické osoby a údajů o zdravotním stavu či o sexuálním životě nebo sexuální orientaci fyzické osoby, obecně zakázáno [5].

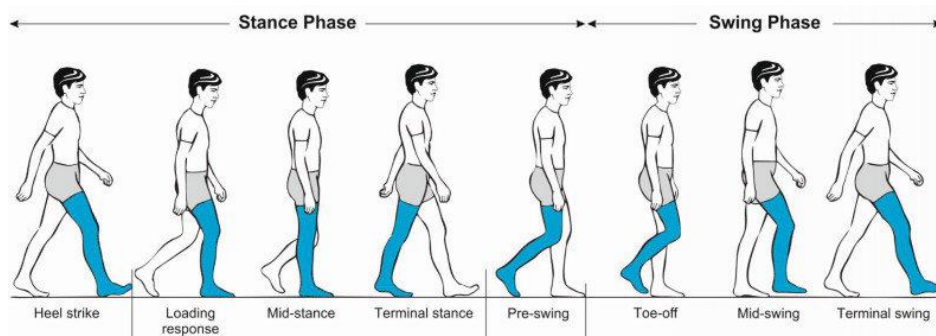
Výjimky jsou uvedeny v odstavci 2 článku 9 GDPR [4]. Z hlediska biometrických systému, a hlavně bezpečnostních a docházkových systému v soukromé sféře je zásadní především výjimka v případě udělení výslovného souhlasu subjektu údajů. Tento souhlas musí být odvolatelný a nesmí být do něj subjekt žádným způsobem nucen.

GDPR se vztahuje ke všem organizacím a subjektům na území evropské unie, které zpracovávají osobní údaje nebo o takovém zpracování rozhodují. Nevztahuje se na čistě soukromé nebo domácí aktivity a použití fyzickými osobami [6].

2.2 Biometrika chůze

Biometrika chůze se zabývá stylem a způsobem chůze. Chůze patří mezi behaviorální biometrické vlastnosti, ovšem často v sobě zahrnuje i fyzické biometrické znaky jako je délka končetin nebo tvar těla. Zdravotnické a psychologické studie našli až 24 komponent chůze použitelných pro rozpoznávání [7]. Například studie unikátnosti kinematiky celého těla G. Park, J. M. Lee a S. Koo během chůze ukázala, že EER ve verifikačním scénáři může dosáhnout 0,27 % a pravděpodobnost unikátní kinematiky chůze v populaci 10 osob tak činní 97,3 % [8]. Pro srovnání otisk prstu dosáhl EER 0,022 % v soutěži verifikace otisků prstu [9].

Chůzi lze rozdělit do opakujících se cyklů. Jeden cyklus lze použít ke zkoumání kompletní chůze. Cyklus se dělí na fáze a člověk během něj udělá dva kroky.



Obrázek 1: Cyklus chůze a jeho fáze [10]

2.2.1 Snímače

K zachycení chůze lze využít široké množství snímačů. Od snímače se do velké míry odvíjí vlastnosti systému, který je používá. Všechny tyto typy snímačů zachycují chůzi v odlišné formě.

Typy snímačů.

- Kamery – Chůzi lze zachytit vizuálně pomocí různých typů kamer. Nejčastěji standardními RGB kamerami, ale používají se i jiné typy jako hloubkové (také nazývané jako TOF kamery). Volba kamery závisí na způsobu reprezentace chůze, podmínkách a ceně. Zaznamenané snímky se po zachycení následně zpracovávají, tak aby z nich byla vytvořena vhodná reprezentace chůze. Právě fáze zpracování může benefitovat z jiného typu kamery než RGB. Velkou výhodou kamery je univerzálnost, dostupnost a možnost zachycení chůze z libovolné vzdálenosti bez jakéhokoliv kontaktu s osobou. Některé typy kamer vyžadují pro svoji funkci příznivé světelné podmínky. Kamera musí mít k dispozici takový prostor, aby dokázala zaznamenat celé tělo a dostatečně dlouhý úsek chůze.
- Akcelerometr – Akcelerometr je zařízení měřící zrychlení, a to posuvné i rotační. Lidé mají během chůze odlišnou akceleraci v 3D prostoru [11]. Akcelerometry bývají často součástí chytrých mobilních telefonů a v důsledku toho jsou rozšířené. Nevýhodou je nutnost nosit akcelerátor u sebe.
- Podlahové senzory – Lidé během chůze na podlahu vyvíjí tlak odlišným způsobem. Tento tlak zachycují podlahové senzory. Použití podlahových senzorů je méně flexibilní než jiné snímače z důvodu větší statičnosti a nutného kontaktu se snímanou plochou [11].
- Radar – Pohyb lidského těla během chůze vytváří Dopplerův jev. Ten zachycuje radar a tato charakteristika chůze je použitelná pro rozpoznávání osob [11].

2.2.2 Vlastnosti

Chůze jako biometrický znak nabízí mnoho unikátních vlastností. Nejširším okruhem těchto vlastností disponuje vizuální forma dat. Mezi největší přednosti se řadí možnost zaznamenání

bezkontaktním a neinvazivním způsobem. V případě kamery a videa z libovolné vzdálenosti. Z videa jako zdroje dat vyplývá další výrazná výhoda, a to je aplikovatelnost na jakýkoliv videozáznam obsahující chůzi. Na rozdíl od biometriky obličeje není vyžadovaná vysoká kvalita záznamu. Dále není nutná specifická orientace těla a je možné i použití v nekontrolovaném prostředí s nespolupracujícími osobami. Uvedené vlastnosti činí kombinaci s dalšími biometrikami, především těmi měkkými, za dobře proveditelnou. Podle studie A. Hadid et al. je spoofing možný, ovšem vyžaduje velké úsilí [12]. Mimo podobný styl chůze vyžaduje také podobnou tělesnou stavbu. V některých systémech může útočníkovi pomoci i podobné oblečení.

Zásadní nevýhodou chůze je její nestálá povaha. Chůzi ovlivňují zdravotní stav, věk, nesení těžkého břemena, některé typy obuvi a další okolnosti. V systémech založených na vizuálním zachycení chůze vzniká problém s oblečením, které zakrývá významné části těla. Velká nevýhodou je také nižší přesnost oproti jiným tradičním metodám jako otisk prstu způsobená uvedenými faktory.

2.2.3 Využití

Biometrika chůze je nová a stále vyvíjející se technologie. Její použití v praxi je v současnosti minimální a málo zdokumentované. Čínská společnost Watrix vytvořila systém pro rozpoznávání osob pomocí chůze [13]. Podle společnosti nabízí přesnost až 96 %. Systém má sloužit k identifikaci osob z videa bezpečnostních kamer. Švédská společnost Wheelys v Číně testuje obchod bez obsluhy využívající biometriku chůze pro zabezpečení [14].

Vlastnosti biometrických systémů využívající chůzi mají velký potenciál pro využití v bezpečnostních systémech a v kriminalistice. Především pak ve zpracování videa z bezpečnostních kamer. Zajímavé může být také použití k pohodlné autentizaci při vstupu do budovy nebo v docházkových systémech. Inteligentní domy mohou v budoucnu využívat chůzi pro spouštění různých personalizovaných akcí. Uživatel by například mohl při příchodu domů spustit požadované rutinní akce jako je příprava kávy nebo nastavení teploty v místnosti.

2.3 Způsoby reprezentace chůze

Přístupy k reprezentaci chůze ve vizuální formě lze rozdělit do dvou hlavních kategorií – model based (založené na modelu) a model free (bez modelu) [11]. Zbylé části práce se budou zabývat převážně chůzí ve vizuální formě.

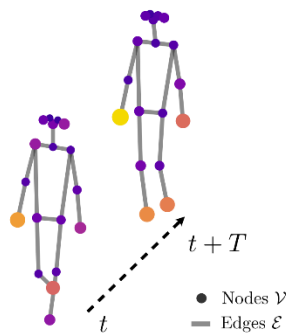
2.3.1 Model based přístupy

Model based přístupy vytváří model chůze a pomocí něj pak chůzi reprezentují. K tomu používají různé parametry jako trajektorie jednotlivých částí těla, délka končetin a tvary různých částí těla. Často se používá kombinace více modelů (například model těla a model pohybu). Tvorba modelu sebou přináší vyšší výpočetní nároky. Výhoda tohoto přístupu je velká nezávislost na úhlu, měřítku a odolnost vůči nežádoucím vlivům. Na druhou stranu zjednodušení a odstranění některých znaků, ke kterému při modelování často dochází, může snížit přesnost. Modely chůze vyžadují vysokou kvalitu videozáznamu, aby mohli přesně extrahovat potřebné informace. [11]

Moderní efektivní nástroje pro odhad postoj umožnili výrazné snížení výpočetní náročnosti a zvýšili popularitu model based přístupů. Oblíbenou reprezentací vytvořenou pomocí klíčových bodů získaných odhadem postoj je skeleton (kostra).

Příklady rozšířených model based reprezentací:

- Skeleton (kostra) – někdy také postoj je definován jako graf $G = (V, E)$, kde V je množina N vrcholů reprezentujících klouby a E je množina hran reprezentujících kosti zachycené maticí sousednosti A [15]. Skeleton se používá v sekvenci. Patří mezi nejrozšířenější model based reprezentace chůze. Vyniká svojí jednoduchostí. Důsledkem jednoduchosti ovšem obsahuje menší množství informací. V současnosti se k získání skeletonu používají moderní systémy odhadu postoj založené na strojovém učení umožňující zpracování v reálném čase. Populárními systémy pro odhad postoj jsou OpenPose [16] a AlphaPose [17]. Skeleton může být v 2D nebo 3D podobě. Používá se samostatně nebo jako součást složitějších modelů.



Obrázek 2: Ukázka reprezentace pomocí Sketetonu [15]

- L. Wang et al. kombinují ve své reprezentaci statické a dynamické vlastnosti těla [18]. Statická část sleduje tvar těla a dynamická úhly kloubů dolních končetin. Dynamická část používá tři modely – model těla, model pohybu a model vazby pohybu.

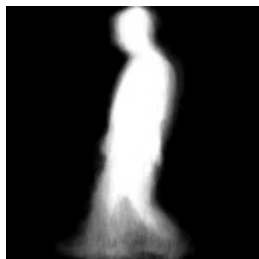
2.3.2 Model free přístupy

Model free přístupy využívají charakteristiky chůze přímo bez vytvoření jakéhokoliv modelu. V praxi to znamená, že jsou snímky videa zpracovány a přímo použity ke klasifikaci. Taková reprezentace obsahuje širší škálu znaků chůze a těla. Přináší sebou ovšem větší množství irelevantních informací a související větší citlivost na anomálie a další nežádoucí jevy. Typicky pracují se siluetami celého těla. Jejich výhodou je nižší výpočetní náročnost a nižší závislost na kvalitě záznamu. Mezi nevýhody patří nižší nezávislost na faktorech jako je oblečení, obuv, velký vliv úhlu kamery a obecně vizuální změny. [11]

Příklady model free reprezentací:

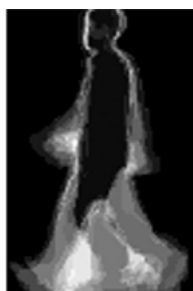
- GEI – Gait energy image (GEI) [19] je významná a velmi populární forma reprezentace chůze. GEI reprezentuje chůzi pomocí jediného obrázku, a to bez ztráty časové informace. Používá k tomu zprůměrování normalizovaných binárních siluet v jednom cyklu chůze. Na pixelech s větší intenzitou se vyskytuje část těla častěji. Oproti sekvenci siluet GEI snižuje výpočetní náročnost, nároky na paměť a snižuje význam šumu. Vznik GEI je umožněn předpoklady – pořadí fází chůze je za běžných

okolností vždy stejné a rozdíly existují právě v těchto fázích. GEI má v oblasti biometrie chůze velký význam a ovlivnil mnoho dalších model free reprezentací.



Obrázek 3: Ukázka GEI [19]

- AEI – Active energy image (AEI) [20] zachycuje chůzi vztahem mezi jednotlivými snímky cyklu chůze. Oproti GEI se více zaměřuje na dynamické oblasti chůze. Vzniká odečtením snímků obsahujících binární siluety s předchozím snímkem a výpočtem aritmetického průměru každého pixelu z vzniklých odečtených siluet. AEI si dává za cíl vyřešit problémy s nízkou kvalitou siluet a vylepšit zachycení dynamických charakteristik chůze. Dle autorů může také snížit vliv oblečení a neseného předmětu.



Obrázek 4: Ukázka AEI [20]

2.4 Tradiční metody

Z důvodu velké komplexnosti, specificky potřeby hledat složité vzory v datech, je strojové učení přirozeným a vhodným způsobem rozpoznávání osob. O tom vypovídá skutečnost, že převážná většina používaných metod patří do strojového učení. Ve výběru jsou rozšířené a významné metody. Vybrané metody nejsou zdaleka jediné.

Do tradičních metod jsou zařazeny metody strojového učení, které byly historicky využívány před metodami hlubokého strojového učení.

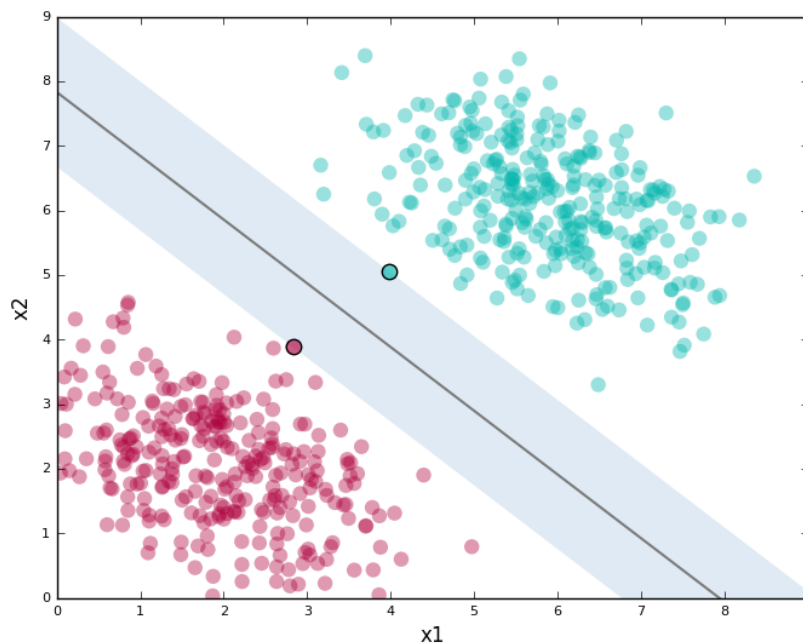
2.4.1 K-Nearest Neighbours (KNN)

Jednoduchý algoritmus strojového učení, který přiřazuje vstupní data do tříd s největší podobností. KNN si během tréninku pouze uloží dataset a přiřazování do tříd probíhá až během klasifikace. Přiřazování vstupu do třídy probíhá výběrem k nejbližších sousedů a následným součtem jejich příslušností do tříd. Vstup je pak přiřazen do třídy, která se mezi sousedy vyskytuje nejčastěji. K výpočtu vzdálenosti datových bodů se používají různé metriky jako Eukleidovská vzdálenost nebo Hammingova vzdálenost. V případě rozhodování na základě pouze jednoho souseda se někdy algoritmu říká pouze Nearest Neighbour (NN) klasifikátor. Je možné se ovšem setkat s vynecháním K i v ekvivalentním významu s KNN (s libovolným k).

S. Sah a S. Prasad Panday použili vážený KNN zohledňující vzdálenost sousedů během rozhodování [21]. V řešeních J. Kovač, V. Štruc a P. Peer [22] a N. Kehtarnavaz et al. [23] slouží KNN jako klasifikátor.

2.4.2 Support Vector Machine (SVM)

Algoritmus strojového učení vhodný především pro klasifikační úlohy, ale použitelný i pro regresi, úlohu predikce spojitych hodnot pomocí zkoumání vztahů příznaků. Úkolem SVM je najít optimální nadrovinu v N -dimenzionálním prostoru, která rozdělí datové body do tříd. Počet dimenzí odpovídá počtu příznaků. Používá k tomu pomocné vektory (support vectors) – datové body, které jsou nejtěžší pro klasifikaci a jsou v nejbližší vzdálenosti k další třídě. Optimální nadrovina se nachází tam, kde jsou hranice (vzdálenosti k pomocným vektorům dvou tříd) největší. Tímto způsobem lze lineárně oddělit dvě třídy. Hledání pomocných vektorů a nadroviny probíhá během trénování. Během výběru pomocných vektorů je nutné ignorovat výrazně atypické datové body (outliers). SVM patří mezi algoritmy strojového učení s učitelem. [28]



Obrázek 5: Lineárně oddělitelné třídy, uprostřed optimální nadrovina, margin v modrém a vyznačené pomocné vektory [24]

V případě lineárně neseparovatelných problémů se používá technika nazývaná jádrový trik. Lineárně neseparovatelný problém se může stát lineárně separovatelným přidáním dalších dimenzí. Jádrový trik umožňuje efektivní mapování dat v původním prostoru do prostoru s větším počtem dimenzí. Využívá toho, že SVM nevyžaduje transformaci všech bodů do více-dimenzionálního prostoru, ale pouze skalární součiny dvojic jednotlivých datových bodů ve více-dimenzionálním prostoru. K tomu slouží jádrové funkce. [24]

Výhodami SVM jsou především nízké požadavky na množství dat a paměťová efektivita. SVM jako klasifikátor ve svém řešení použili X. Wang, J. Wang a K. Yan [25], H. Masood, H. Farooq a M. T. Mahmood [26] a D. Das a S. Saharia [27].

2.6 Hluboké učení

Hluboké učení (deep learning) je forma strojového učení používající neuronové sítě s velkým počtem vrstev. Velký počet vrstev umožňuje efektivní hledání komplexní příznaků v datech. Výhodou hlubokého učení je automatizování extrakce příznaků vyplývajících z více vrstev.

Neuronové sítě jsou systémy inspirované lidským mozkem skládající se z vrstev neuronů. Neuron označuje v rámci neuronové sítě uzel, který má vážené vstupy a výstup udávaný aktivační funkcí. Populárními aktivačními funkcemi jsou ReLu, Tanh a Sigmoida. K vstupům bývá přičtena konstanta nazývaná bias způsobující posun aktivačních funkcí.

Neurony jsou skládány do vrstev. Vrstvy se dělí na vstupní, skryté a výstupní. Vstupní vrstva slouží k načtení dat pro další vrstvy. Skryté vrstvy se nachází mezi vstupní a výstupní vrstvou. Transformují vstup a hledají příznaků vhodné pro určení výstupu. Více skrytých vrstev odlišuje hluboké nerunové sítě od běžných umělých neuronových sítí. Výstupní vrstva připraví výstup v požadované podobě. Vrstvy mohou být různého typu. Například hustě propojená vrstva je vrstva neuronů, kde každý neuron vrstvy je propojený se všemi výstupy předešlé vrstvy. Vstupy neuronu jsou zpracovány skalárním součinem s vahami a následným součtem s biasem.

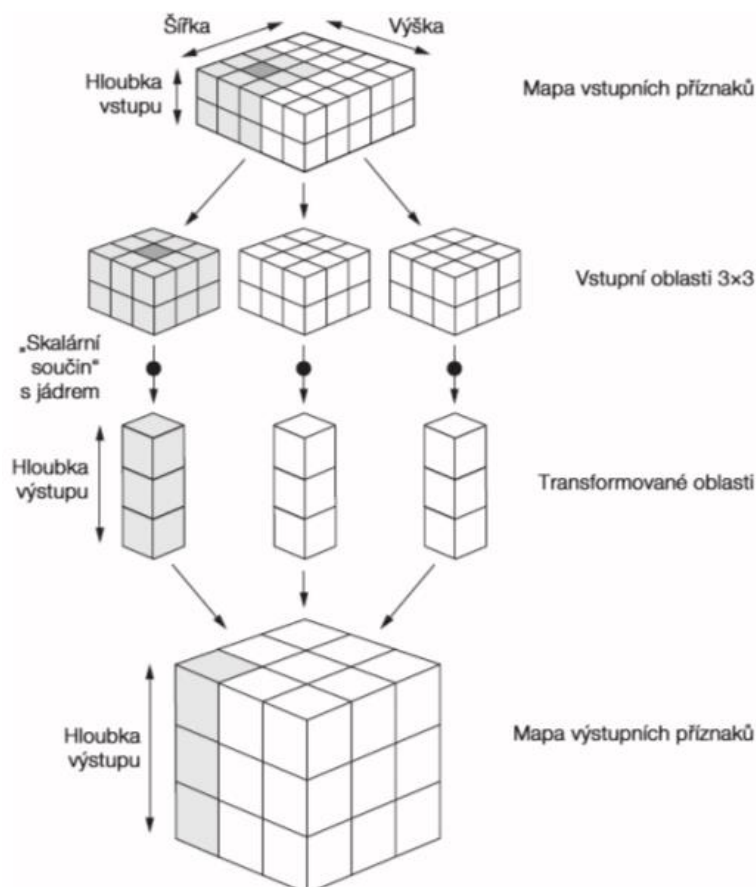
Během učení dochází k optimalizaci vstupních vah neuronů. Chyba je definována chybovou funkcí. Na základě chyby jsou váhy a biasy upravovány optimalizátorem, který usiluje o nalezení optimálních hodnot vah s nejnižší výslednou chybou. Většina optimalizátorů k tomu využívá algoritmus zpětného šíření chyby, který efektivně počítá gradient chybové funkce v závislosti na vahách. Gradient označuje vektorové pole, jehož hodnota v bodě p je vektor složený z parciálních derivací funkce f v p . Lze ho interpretovat jako směr a velikost největšího nárůstu. Učení probíhá po epochách. Epocha označuje průchod všemi příklady trénovací množiny dat. Počet zpracovaných příkladů, po kterých dojde k úpravě parametrů sítě, se nazývá dávka. Během inference označuje dávka data zpracovaná společně. Inference označuje proces provedení predikce modelem.

2.6.1 Konvoluční neuronové síť

Konvoluční neuronové síť (CNN) jsou populární typ hlubokých neuronových sítí využívající konvoluční jádra (filtry). CNN pracují s mapami příznaků – tensorů s dimenzemi výška, šířka a osa kanálů. Ve vstupní vrstvě je mapou příznaků obrázek a kanály jsou nejčastěji kanály RGB barev jednotlivých pixelů. V dalších vrstvách kanály odpovídají jednotlivým filtrům. [28]

Konvoluční jádra jsou matice obsahující naučené váhy sloužící k extrakci příznaků typicky v prostoru 3×3 , 5×5 nebo 7×7 . Přikládají se ke každému místu na vstupní mapě příznaků a pomocí součinu tensorů transformují oblast do 1D vektoru s tvarem osy kanálu výstupu, který umístí na místo ve výstupní mapě odpovídající přiložené pozici ve vstupní mapě. Aby bylo možné přiložit konvoluční jádro k okraji a dosáhnout stejných rozměrů vstupu a výstupu, tak lze potřebné oblasti doplnit o výplň (padding). Počet pozic posunutí jádra se nazývá kroky (strides). Nejběžněji se používá krok velikosti 1. Jiné kroky snižují rozměry výstupní mapy příznaků. První konvoluční vrstvy se učí jednoduché tvary jako jsou hrany a další v nich

hledají složitější vzory jako například oči, nos atd. a postupně se z jednodušších vzorů skládají složitější. Tímto způsobem se CNN dokáží učit komplexní abstraktní vzory. [28]



Obrázek 6: Průběh činnosti konvoluční vrstvy [28]

Kromě konvolučních vrstev se v CNN často používají pooling vrstvy. Pooling vrstvy zmenšují rozměry mapy příznaků. Podobně jako u konvolučních jader se po vstupní mapě příznaků posouvá okno. Ovšem s krokem odpovídajícím velikosti okna a výstupem určeným maximem nebo průměrem oblasti okna. Mimo zmenšení rozměrů mapy příznaků umožňuje pooling filtrům zpracovávat větší oblast z původního vstupu a přidává invarianci příznaků proti drobným posunům a změnám. Na konci CNN se nachází hustě propojené vrstvy. Konvoluční neuronová síť, která obsahuje pouze konvoluční vrstvy, se nazývá plně konvoluční síť. [28]

Velkou výhodou konvolučních neuronových sítí je jejich schopnost učit se vzory bez ohledu na jejich polohu. Konvoluční vrstvy mají menší počet parametrů než hustě propojené vrstvy a jsou výrazně výpočetně efektivnější. Způsob, jakým se učí vizuální vzory od nejjednodušších, ze kterých postupně skládají komplexnější, je přirozený a velmi efektivní.

Zajímavá je také jejich schopnost učit se hierarchii vzorů a reprezentovat komplexní vizuální pojmy, která z nich dělá efektivní způsob zpracování vizuálních dat.

V rámci biometriky chůze se pro svoji schopnost pracovat s vizuálními daty často používají pro zpracování model free reprezentací chůze. K. Shiraga et al. vytvořil ve svém řešení CNN zpracovávající GEI s výstupní softmax vrstvou a výsledkem reprezentujícím podobnost k jednotlivým identitám [29]. C. Li et al. [30] použili CNN pro extrakci příznaků a Joint Bayesian metodu [31] pro určení podobnosti. A. Sokolova a A. Konushin přišli s řešením, které používá optický tok páru snímků chůze limitovaný na nejvýznamnější části těla [32]. CNN v tomto řešení extrahuje příznaky a ty se používají pro klasifikaci pomocí NN klasifikátoru.

2.6.2 Siamské neuronové sítě

Siamské neuronové sítě jsou typ sítí, který obsahuje více identických podsítí ze svými vlastními vstupy, ale společnými a stejnými váhami. Používají se především pro porovnávání dvou vstupů. Obvykle je z výstupů obou podsítí vypočtena Euklidovská vzdálenost nebo jiná metrika vyjadřující podobnost vstupů sítě.

Euklidovská vzdálenost v N -dimenzionálním prostoru je definována rovnicí:

$$D(p, q) = \sqrt{\sum_{i=1}^N (p_i - q_i)^2}$$

Kde p a q jsou body v N -dimenzionálním prostoru.

K učení se používají chybové funkce contrastive loss nebo triplet loss. Obě funkce pracují se vzdálenostmi vstupů a skutečností, jestli vstupy patří do stejné třídy.

Contrastive loss [33] se snaží dosáhnout toho, aby byly datové body ze stejné třídy umístěny do určité vzdálenosti (margin) a všechny body mimo tuto třídy měly vzdálenost větší. Z důvodu lepší generalizace se nesnaží přitahovat body z totožné třídy a odtahovat ty z odlišné více než je nutné. Generalizace označuje schopnost sítě klasifikovat neznámé příklady.

Definice původní contrastive loss funkce [33]:

$$L(W, Y, \vec{X}_1, \vec{X}_2) = (1 - Y) \frac{1}{2} (D_w)^2 + (Y) \frac{1}{2} \{\max(0, m - D_w)\}^2$$

- W – naučené váhy
- $Y = 0$ pokud vstupy patří stejné osobě a $Y = 1$ pokud jiné
- \vec{X}_1, \vec{X}_2 – vstupní vektory
- D_w – zkrácený zápis pro Euklidovu vzdálenost $D_w(\vec{X}_1, \vec{X}_2)$, která je počítaná z výstupů $G_w(\vec{X}_1)$ a $G_w(\vec{X}_2)$
- G_w – učená parametrická funkce mapující vysoko-dimenzionální vstup na varietu, tak aby Euklidovská vzdálenost dvou jejich výstupů přibližně odpovídala sémantické podobnosti vstupů
- $m > 0$ – margin

Triplet loss [34] na rozdíl od contrastive loss vyžaduje použití sítě s třemi vstupy pro zkoumaný vstup, příklad ze stejné třídy a příklad z odlišné třídy. Základem triplet loss je myšlenka, že členové třídy mají vzájemně velmi podobné znaky, které jsou odlišnější od členů jiných tříd. V průběhu učení se snaží zmenšovat vzdálenost k příkladu ze stejné třídy a zvyšovat tu k příkladu z odlišné třídy. Odlišný pár musí být od stejného páru vzdálený určité vzdáleností, kterou udává margin.

Definice původní triplet loss funkce N trojic [34]:

$$L = \sum_i^N D(f(x_i^a), f(x_i^p))^2 - D(f(x_i^a), f(x_i^n))^2 + \alpha$$

- $f(x)$ – funkce mapující vstup na embedding – nízko-dimenzionální vektor sémanticky zachycující vstup
- N – celkový počet trojic
- a – zkoumaný vstup (anchor)
- p – pozitivní příklad

- n – negativní příklad
- α – margin

Siamské neuronové sítě jsou vhodné pro verifikaci a identifikaci. Mimo chůze byly aplikovány v rámci rozpoznávání pomocí obličeje. Z těch důvodů jsou zajímavé pro biometriku chůze.

N. Takemura et al. [35] vytvořili dvě siamské neuronové sítě rozpoznávající chůze pracující s GEI. Jedna používá contrastive loss a druhá triplet loss. Mimo nich ještě vytvořil dvě konvoluční sítě s rozdílem dvou GEI na vstupu a totožnými chybovými funkcemi.

2.6.3 Rekurentní neuronové sítě

Rekurentní neuronové sítě (RNN) jsou typ neuronových sítí pracující se sekvencemi a časovými řadami. Mezi jejich nejvýznamnější vlastnosti patří, že mají paměť. Postupně iterují vstup a pamatují si předešlá data. RNN kombinují vstup v čase s předešlým stavem a výstup nastaví jako nový stav. Tento proces probíhá v rekurentních vrstvách. Ty mohou vrátit konečný výstup nebo sekvenci výstupů pro každý časový bod. [28]

Z důvodu velkého množství příznaků a problému mizení gradientu se pro rozpoznávání chůze používají především dva konkrétní typy rekurentních neuronových sítí – LSTM a GRU sítě. Problém s mizením gradientu označuje situaci, kdy během zpětného šíření chyby gradient v prvních vrstvách neuronové sítě nabývá hodnot blízkých nule a vrstvy se neučí [36].

LSTM (Long Short-Term Memory) přidává další datový tok, který obsahuje dlouhodobé informace a zamezuje mizení gradientu. Při získávání další hodnoty datového toku se provádí tři transformace vstupu a stavu, jejichž váhy se síť učí. [28]

LSTM Transformace [28]:

$$i_t = \text{activation}(s_t \cdot U_i + x_t \cdot W_i + b_i)$$

$$f_t = \text{activation}(s_t \cdot U_f + x_t \cdot W_f + b_f)$$

$$k_t = \text{activation}(s_t \cdot U_k + x_t \cdot W_k + b_k)$$

- *activation* – aktivační funkce

- s_t – stav v čase t
- x_t – vstup v čase t
- U – matice vah stavu
- W – matice vah vstupu
- b – bias

Výstup LSTM toku dat v čase $n + 1$ [28]:

$$c_{t+1} = i_t * k_t + c_t * f_t$$

GRU (Gated Recurrent Unit) funguje podobně jako LSTM. Odlišuje se především počtem prováděných transformací, kde GRU provádí pouze dvě. Nabízí tak podobnou funkcionalitu a výsledky s menší výpočetní náročností. GRU je vhodnější hlavně pro kratší sekvence. [28]

Použití rekurentních neuronových sítí se pro rozpoznání pomocí chůze nabízí vzhledem k faktu, že chůze je sekvence a jednotlivé pohyby na sebe navazují. Jejich hlavní výhoda spočívá v zachování veškerých časoprostorových informací. X. Wang a W. Yan navrhli Conv LSTM síť úspěšně zpracovávající sekvenci snímků ve formě ff-GEI – formát upravující GEI, tak aby do každého snímku přidával historickou informaci [37]. Conv LSTM je varianta LSTM používající konvoluci v rámci transformací LSTM datového toku. LSTM sítě jsou populární i pro rozpoznání založené na jiných senzorech než kamera.

2.6.4 Autoenkodéry

Autoenkodéry jsou typ generativních neuronových sítí, které slouží ke kódování vstupu, redukci dat a odstranění šumu. Skládají se ze dvou hlavních částí – kodér a dekodér. Kodér mapuje vstup do typicky menšího vektorového prostoru a dekodér se snaží rekonstruovat kódovaná data zpátky do původní formy. [39]

S. Yu et al. pomocí autoenkodéru řešili problém vlivu rozdílných úhlů, oblečení a různých nesených předmětů [40]. Použili k tomu několik autoenkodérů transformujících GEI. Autoenkodéry jsou seřazeny za sebou ve vrstvách, kde každá vrstva dokáže provést změnu

úhlu o 18° a odstranění vlivu oblečení. Cílem je dosáhnout GEI bez nežádoucích vlivů v bočním pohledu. Příznaky cílové reprezentace se pak používají pro klasifikaci.

2.6.5 Generativní soupeřící síť

Generativní soupeřící síť (GAN) jsou primárně zaměřené na generování kvalitních syntetických obrázků. Skládají se ze dvou částí – generátoru a diskriminátoru. Úkolem generátoru je vytvářet umělé obrázky a diskriminátor musí ze směsi umělých a reálných obrázků určit, zda jsou skutečné nebo umělé. Obě části tak mají protichůdné cíle a soupeří spolu. V průběhu souboje se postupně zdokonalují. [28]

B. Hu et al. zkombinovali generativní soupeřící síť s autoenkodérem a vytvořili model, který dokáže převést GEI do libovolného úhlu [41]. GAN trénuje autoenkodér tak, aby kodér dokázal získat ze vstupního GEI příznaky nezávislé na úhlu pohledu a dekodér z nich a zvoleného úhlu vytvořil GEI. Používá dva diskriminátory – úhlu a identity.

2.6.6 Kapslové síť

Kapslové síť vznikly, aby zdokonalily modelování hierarchických vztahů příznaků konvolučních neuronových sítí. Konvoluční neuronové síť jsou náchylné na chyby v prostorových vztazích jednotlivých vzorů. To znamená, že CNN rozpozná objekt i v případě, kdy jsou jeho jednotlivé části ve špatné vzájemné prostorové pozici. Mimo to může mít CNN problém s různými transformacemi. Kapslové síť se snaží tyto problémy vyřešit a nabídnout lepší reprezentaci příznaků pomocí tzv. kapslí. Kapsle je skupina neuronů, kde každý neuron zachycuje určitou vlastnost stejného příznaku. Na vstupech a výstupu mají vektor. To jim umožňuje učit se kromě příznaků samotných také jejich deformace a rozdílné úhly pohledu. Vstupem kapsle jsou příznaky z CNN. [38]

Z. Xu et al. vytvořili dvě kapslové neuronové síť pro rozpoznávání chůze s cílem vylepšit výsledky konvolučních neuronových sítí [42]. Obě síť se skládají z konvoluční části a kapslové. Konvoluční část provádí detekci lokálních příznaků a kapslová se snaží vytvořit z nich lepší reprezentaci.

2.5 Podobné existující aplikace

Veřejné projekty podobné systému, který je předmětem této práce, existují výhradně ve formě projektů na stránce pro hostování kódu GitHub. Martin Persson se ve své diplomové práci také zabýval tvorbou automatického systému [43]. Chůzi reprezentoval skeletem a rozpoznávání prováděl pomocí LSTM sítě. Jeho systém je méně kompletní, neřeší detaily související s praktickým použitím a neusiluje o dosažení stejných cílů jako navržený systém. Automated-Gait-Analysis [44], DNN for gait recognition in TensorFlow [45], Human Identification Using Openpose [46] a Skeleton-based Gait Recognition via Robust Frame-level Matching [47] jsou automatické systémy používající model based reprezentaci, které jsou formou, rozsahem a zaměřením totožné s Perssonovým systémem. Většina projektů zaměřených na chůzi jsou pouze implementace modelů pro rozpoznávání nebo jednoduché systémy, které bez úprav a rozšíření nejsou vhodné pro praktické použití. Podobný veřejně dostupný framework ani systém, který by nabízel kompletní prakticky použitelné řešení, v době vzniku práce nebyl nalezen.

OpenGait framework [48] nabízí prostředí pro vývoj a testování vlastních modelů a jejich porovnávání s relevantními existujícími modely pro rozpoznávání pomocí chůze. Systém navržený v této práci je orientovaný naprosto jiným směrem, ale obsahuje také některé nástroje usnadňující vývoj vlastního modelu a umožňuje učení modelů poskytnutých v balíčku se systémem.

3 Praktická část

Tato část práce se bude zabývat navrženým systémem pro automatické rozpoznávání osob pomocí chůze. Na začátku bude popsán celý systém a následně pak detailně činnost jednotlivých částí.

Rozpoznání osob bude vždy probíhat z bočního pohledu v úhlech v rozmezí 54 až 126 stupňů. Systém je zaměřený na možnost provozu na širokém spektru běžných nevykonných osobních zařízeních bez specializovaných součástí. Takové zařízení bude reprezentováno notebookem bez dedikované grafické karty s procesorem Intel i3-8130U (2018). Tento notebooku bude sloužit jako měřítko a systém na něm bude laděný a testovaný. Jedním z cílů je provádění rozpoznání na uvedeném zařízení v reálném čase. To znamená, že systém musí dostatečně rychle zpracovávat snímky a provádět klasifikaci. K splnění cíle a dosažení dostatečné klasifikační přesnosti budou použity moderní metody a technologie. Vzhledem k orientaci na praktické použití musí systém fungovat v různých prostředích a podmínkách. Během vývoje byl kladen důraz na modulárnost, aby bylo možné použít jednotlivé části mimo systém nebo je nahradit vlastními. Součástí řešení jsou vlastní implementace modelů a podpůrné skripty, které slouží k předzpracování dat, trénování modelů a jejich testování.

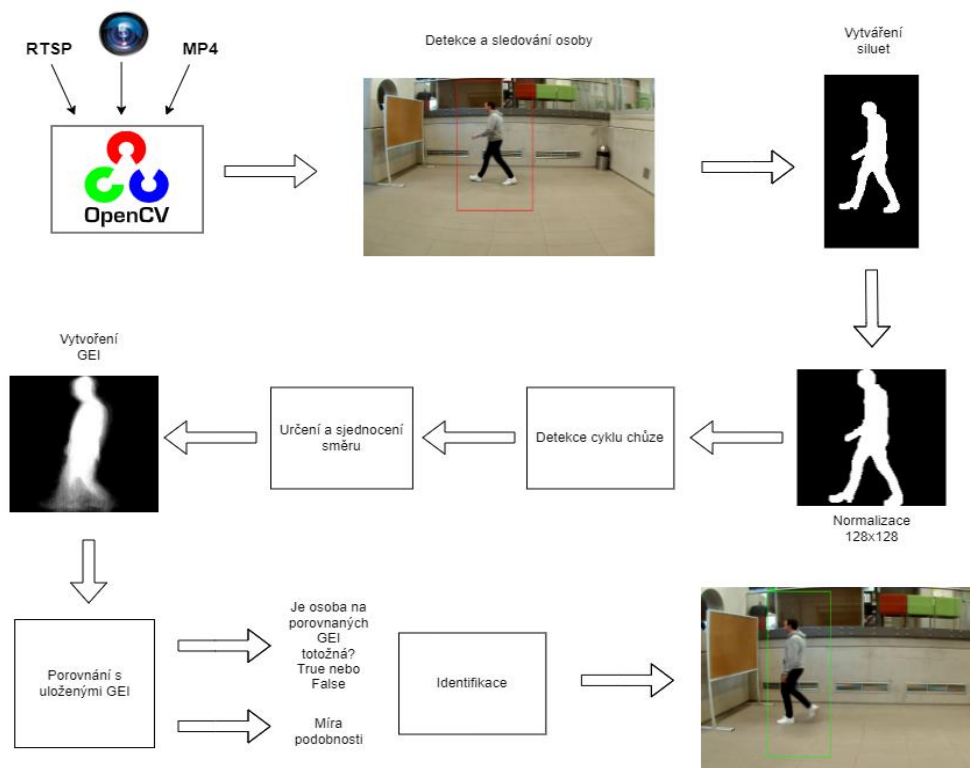
3.1 Popis řešení

Před samotným začátkem rozpoznávání se do paměti načtou uživatelé a šablony jejich chůze ve formátu GEI.

Proces rozpoznání začíná v systému načítáním videa. Načítání probíhá v odlišném vlákně asynchronním způsobem. Asynchronní načítání ve vlákně uvolňuje procesor během načítání snímků. Navrhovaný systém používá cPython, nejrozšířenější implementaci Pythonu, která odlišným vláknům neumožňuje pracovat s jinými jádry procesoru. Video načítá OpenCV, které podporuje široké množství vstupních formátů. Velikost vyrovnávací paměti snímků (OpenCV VideoCapture buffer) byla nastavena na 3, aby u videa nedocházelo ke zpožděním a byl zachován chod v reálném čase.

Po načtení snímku probíhá detekce osob. Během detekce jsou všechny osoby na snímku označeny pomocí bounding boxů. Bounding box je obdélník ohraničující oblast, kde se nachází detekovaný objekt.

Bounding boxy patřící ke stejnému objektu musí být k němu napříč snímky přiřazeny. Tento problém řeší sledování objektů ve video (object tracking). Při první detekci objektu vznikne stopa, ke které jsou následně přiřazovány detekce v dalších snímcích.



Obrázek 7: Proces identifikace

Oblasti ohraničené bounding boxy slouží k vytváření siluet. Výhodou vytváření siluet pouze z těchto oblastí je nižší výpočetní náročnost, velikost objektů poměrně k obrázku a skutečnost, že u menších objektů nedochází ke ztrátě informací, protože při jejich zpracování nemusí docházet ke zmenšení. To přináší lepší kvalitu vytvořených siluet a do určitého množství objektů lepší rychlost.

Vzniklé siluety je nutné normalizovat do uniformních rozměrů a vycentrovat. Nejprve dochází k odstranění nežádoucího šumu na obrázku siluety. K tomu je využita morfologická operace otevření s jádrem rozměrů 5x5. Morfologické operace jsou jednoduché operace s objekty v obrázku (obvykle binárním) [49]. Jádro se posouvá po obrázku a určuje hodnoty jednotlivých pixelů. Použitá operace otevření odstraňuje drobné shluky bílých pixelů. Dále byla silueta oříznuta o veškeré nadbytečné pixely. Vycentrování bylo dosaženo výpočtem mediánu souřadnic šířky všech pixelů siluety a doplnění rozměrů obrázku prázdnými pixely, tak aby vzdálenost od okraje k mediánu byla z obou stran stejná. Následuje rovnoměrné doplnění všech stran obrázku do čtverce a změna rozměrů na 128x128. Během výběru rozměrů byla zohledněna rychlost zpracování (u menšího obrázku probíhá méně výpočtů),

dostatek horizontálního prostoru (nedostatek naruší normalizaci), velikost získaných siluet (určuje velikost, od které dochází ke ztrátě informací) a přesnost (ztráta informací během zmenšení).

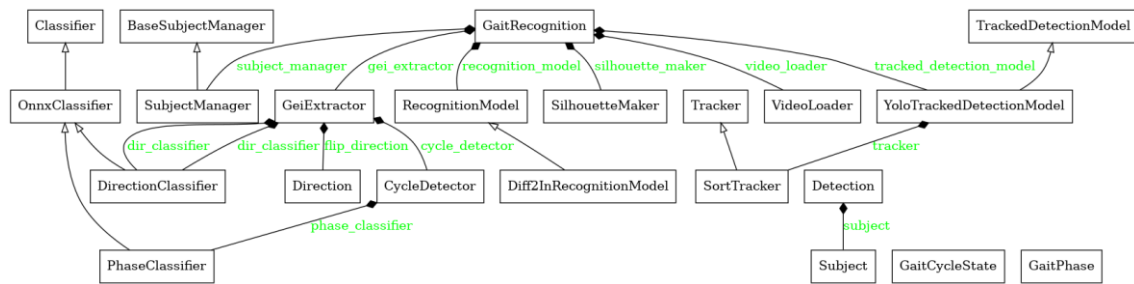
System pracuje s chůzí reprezentovanou pomocí GEI. Pro vytvoření této reprezentace je nutné zachytit začátek a konec cyklu chůze. Tím se zároveň vyřeší zachycení chůze jako takové. Zprůměrování hodnot každého pixelu napříč sekvencí siluet cyklu vznikne GEI. Současně s detekováním cyklu chůze dochází k určení směru, kterým chůze probíhá a GEI se následně nasměruje sjednoceným směrem. Sjednocený směr výrazně usnadňuje klasifikaci a zlepšuje tak úspěšnost.

Po vytvoření GEI probíhá klasifikace. Klasifikace probíhá asynchronně v odlišném vláknu. Důvodem k použití vláken místo procesů je, že klasifikaci provádí model strojového učení pomocí frameworku, který již využívá více jader procesoru. Multiprocessing, jež umožňuje v Pythonu využití více jader procesoru, by proto nepřinesl žádné výhody, a naopak přidal by nadbytečnou režii s ním spojenou. V případě použití grafické karty provádí klasifikaci procesor a grafická karta se stará o ostatní činnosti, aby si zpracování videa zachovalo plynulost. Při klasifikaci model porovnává GEI rozpoznávané osoby a všech známých uživatelů v systému. Na výstup dodává skóre odlišnosti, kde vyšší skóre značí větší odlišnost mezi porovnávanými GEI. O tom, jestli GEI patří stejné osobě rozhoduje práh zvolený pomocí ROC křivky. Klasifikace probíhá v dávkách s výchozí velikostí 5, protože inference v dávkách je efektivnější. Identifikace probíhá po dokončení všech klasifikací. V případě více pozitivně vyhodnocených uživatelů je osoba identifikována jako ten z nejmenším skóre odlišnosti.

Pro zajištění spolehlivého chodu na nevykonných zařízeních probíhá rozpoznávání v každé chvíli pouze pro jednu osobu. Osoby se mění po dokončení identifikace, pokud identifikace selže nebo pokud je identifikovaná osoba po určitou dobu nečinná v zahajovací fázi.

Výchozí aplikace umožňuje importovat uživatele a následně provádět jejich detekci a identifikaci. Detekce uživatelů se ukládají do minimalistické databáze.

3.2 Struktura systému



Obrázek 8: Zjednodušený UML diagram systému

- GaitRecognition – Třída spojující veškeré části do jednoho celku. Je zodpovědná za import uživatelů a jejich identifikaci.
- SubjectManager – Třída zodpovědná za správu subjektů a získávání jejich dat.
- DirectionClassifier – Třída zodpovědná za klasifikaci směru. Obaluje model pro klasifikaci směru.
- PhaseClassifier – Třída zodpovědná za klasifikaci fáze kroku. Obaluje model pro klasifikaci fáze kroku.
- CycleDetector – Třída detekující a zaznamenávající cyklus chůze.
- GeiExtractor – Třída zodpovědná za vytváření GEI ze sekvence snímků.
- RecognitionModel – Třída provádějící rozpoznání chůze.
- SilhouetteMaker – Třída pro vytváření siluet.
- VideoLoader – Třída zodpovědná za načítání videa.
- SortTracker – Třída implementující SORT algoritmus pro sledování objektů.
- YoloTrackedDetectionModel – Třída kombinující detekci a sledování objektů.

3.3 Použité technologie

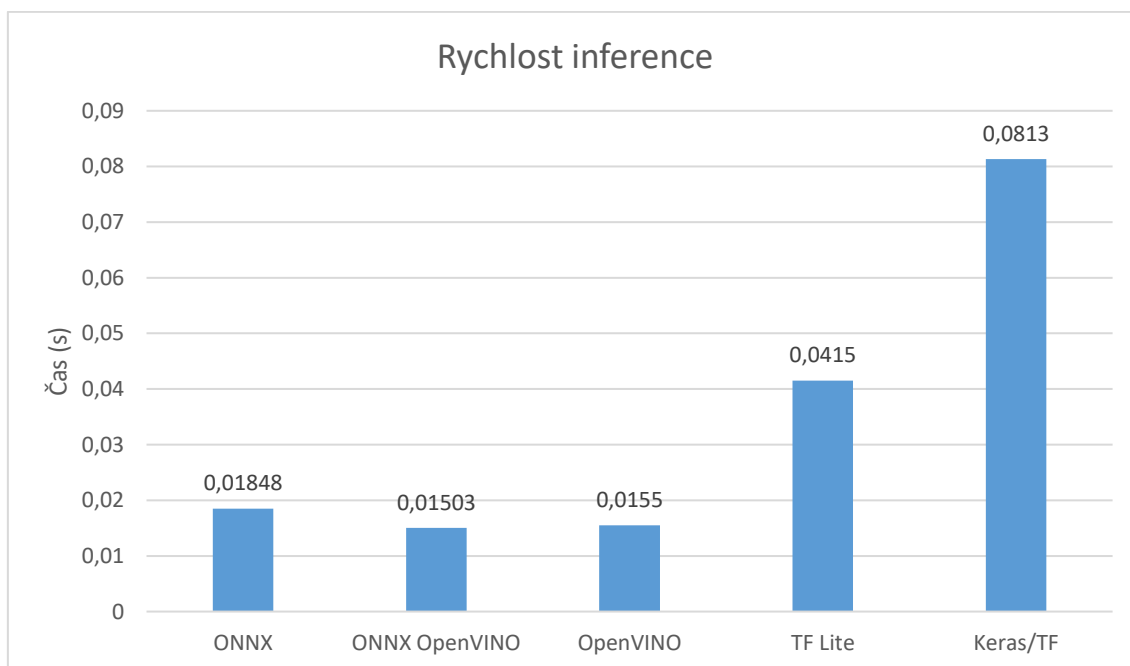
System je implementovaný v programovacím jazyce Python. Primárním důvodem je velká podpora knihoven a nástrojů pro strojové učení a strojové vidění. Mezi další výhody patří jeho popularita, pohodlné použití a dobrá podpora různých platforem. Nevýhodou je nízká rychlost, která je ovšem kompenzována velkým množstvím kvalitních knihoven napsaných v jazyce C.

Veškeré vlastní implementace modelů jsou vytvořeny v Kerasu. Keras je vysoko-úrovňové rozhraní pro strojové učení. V minulosti se jednalo o samostatný projekt, který umožňoval použití nad různými frameworky. Momentálně je součástí frameworku Tensorflow. Jeho

hlavní výhodou je jednoduché a efektivní používání. Vzhledem k integraci ve frameworku Tensorflow lze Keras v případě potřeby libovolně kombinovat s nízko-úrovňovým rozhraním.

Z důvodu zaměření systému bylo nutné vybrat efektivní prostředí pro inferenci modelů. Porovnání proběhlo mezi prostředím Tensorflow, TFLite (Tensorflow Lite), ONNX (Open Neural Network Exchange) a OpenVINO. Tensorflow je výchozí prostředí. Pro ostatní prostředí bylo nutné model převést do podporovaného formátu. TFLite nabízí soubor nástrojů zaměřených na mobilní a edge zařízení. ONNX je otevřený formát pro modely strojového učení, který umožňuje oboustranný převod s ostatními formáty. ONNX ekosystému obsahuje runtime, který automaticky nachází příležitosti pro optimalizaci a poskytuje hardwarovou akceleraci pro širokou škálu zařízení. OpenVINO představuje soubor nástrojů pro optimalizaci modelů na zařízeních od společnosti Intel. ONNX umožňuje využívat OpenVINO pro hardwarovou akceleraci. Tato varianta byla také zahrnuta do testování.

Testování proběhlo na modelu pro vytváření siluet. Vzhledem k požadovanému zpracování dat v reálném času byla testována inferencí po jednom obrázku. Model byl na všech platformách testován po základní optimalizaci, tj. model byl pouze převeden do správného formátu, nastaven a spuštěn. U ONNX bylo za základní optimalizaci považováno plné nastavení grafové optimalizace.



Graf 1 – Výsledné srovnání rychlosti inference

Nejlépe si v testu vedly platformy OpenVINO a ONNX. Nakonec byl vybrán formát ONNX. Výraznou nevýhodou OpenVINO je omezení na zařízení od Intelu. V kontrastu ONNX podporuje široké spektrum zařízení včetně grafických karet Nvidia a dokonce umožňuje využívat OpenVINO v pozadí. Dále lze modely pro ONNX vytvářet v libovolném frameworku. To poskytuje systému velkou flexibilitu. Vzhledem k ambicím ONNX stát se společným formátem pro strojové učení a open-source povaze lze očekávat, že tato vlastnost bude do budoucna zachována.

Pro detekci osob byl vybrán model YOLOv5, který je implementovaný ve frameworku PyTorch. Použita je implementace přímo od autorů. Bližší informace o modelu lze nalézt v kapitole o detekci osob v obraze.

Knihovna Albumentations poskytuje způsoby augmentace dat. Obraz je zpracováván pomocí knihovny pro počítačové vidění OpenCV. Pro operace s multidimenzionálními poli je využívána knihovna Numpy. Systém dále používá Kálmánův filtr z knihovny FilterPy a funkci pro vytváření ROC křivek z knihovny pro vědecké výpočty scipy. Konfigurační soubor má formát YAML a o jeho načítání se stará knihovna PyYAML. YAML byl vybrán pro svou eleganci a dobrou čitelnost.

Využívání databáze probíhá přes peewee. Peewee je jednoduché a lehké ORM s podporou SQLite, MySQL, PostgreSQL a CockroachDB. SQLite bylo vybráno pro výchozí databázi. Důvody pro výběr jsou přenosnost, jednoduchost, malá velikost a žádná instalace ani nastavování. Tyto vlastnosti byly při výběru výchozí databáze prioritizovány.

3.4 Datasetsy

Datasetsy použité pro trénování modelů strojového učení:

- COCO [50] – Rozsáhlý dataset vydaný Microsoftem, který obsahuje obrázky s objekty různých tříd pro detekci, sémantickou a instanční segmentaci a mnoho dalších úloh. Často se používá pro porovnávání modelů. V této práci byl konkrétně použit dataset COCO Detections 2017. Obrázky byly zredukovány na ty, které obsahují osoby. Převod na binární masky byl proveden s pomocí oficiální Python API datasetu. Obrázky, kde se nachází pouze malé části osob, byly také vyfiltrovány. Výsledná trénovací množina obsahuje 32 629 obrázků a validační obsahuje 1 337 obrázků.

- Segmentation Full Body TickTok Dancing Dataset [51] – Dataset tančících osob ze sociální sítě TickTok. Obsahuje 2615 obrázků s binárními maskami.
- Penn-Fudan Dataset for Pedestrian Detection [52] – Dataset pořízený na univerzitách University of Pennsylvania a Fudan University. Skládá se ze 170 obrázků, na kterých se nachází 345 chodců zachycených na ulici. Instance chodců jsou od sebe vzájemně odlišeny.
- CASIA A [53] – První z datasetů CASIA. CASIA označuje Chinese Academy of Sciences, The Institute of Automation. Institut zodpovědný za tvorbu datasetu. Dataset obsahuje 12 sekvencí od 20 osob. Sekvence jsou ve třech úhlech 0° , 45° a 90° . Ke každému úhlu jsou od každé osoby 4 sekvence. K dispozici jsou binární siluety.
- CASIA B [54] [55] – Je nejrozsáhlejší z CASIA datasetů. Dataset tvoří záznamy chůze 124 osob v 11 úhlech. Sekvence vždy existují ve třech variantách běžné oblečení, zimní oblečení a s batohem/taškou. K dispozici jsou binární siluety. Siluety mají větší kvalitu než ty v CASIA A.
- CASIA C [56][55] – Byl pořízen infračervenou kamerou v noci a obsahuje 153 subjektů ve 4 podmínkách – normální chůze, pomalá chůze, rychlá chůze a chůze s batohem. K dispozici jsou binární siluety.
- OU-MVLP (The OU-ISIR Gait Database, Multi-View Large Population Dataset) [57] – Obsahuje siluety a GEI od 10 307 subjektů v úhlech od 0 do 270 stupňů. Pro každý subjekt a úhel obsahuje 2 průchody. Subjekty mají vždy stejné oblečení. Dataset vznikl v Institutu věd a průmyslového výzkumu Ósacké univerzity.
- Persons Labeled [58] – Dataset fotek lidí s přesnými anotacemi jednotlivých instancí. V práci byla použita verze s vyfiltrovanými fotkami s nekvalitním značením, která obsahuje binární siluety [59]. Počet obrázků v této verzi je 2667.

3.5 Části systému

V této části bude čtenáři představen detailní popis činnosti jednotlivých částí systému. Budou zde popsány používané metody, jejich výběr a implementace.

3.5.1 Detekce objektů

Detekce objektů označuje proces hledání pozice objektů patřících do určitých tříd v obraze. Výstupem jsou typicky bounding boxy. Detekce objektů není v systému řešena vlastní implementací a v této části je popsána analýza metod a výběr modelu.

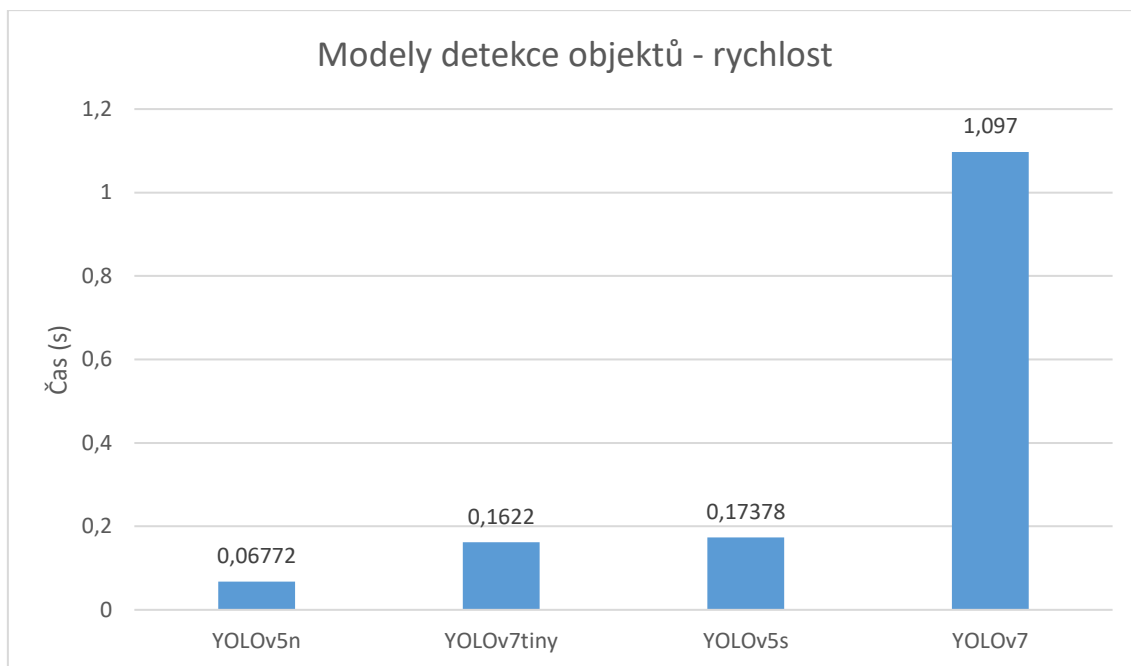
Moderní algoritmy pro detekci objektů se dělí do dvou kategorií. První kategorii tvoří algoritmy založené na návrhu regionů [60]. Detekce probíhá ve dvou krocích. V první fázi jsou navrženy regiony, kde by se objekty mohly vyskytovat. Následně probíhá klasifikace výskytu objektů z různých tříd v jednotlivých regionech. Do této kategorie patří algoritmy R - CNN, Fast R -CNN a Faster-RCNN. Velkou nevýhodou je nízká rychlost, která zamezuje zpracování v reálném čase. Z toho důvodu nebylo uvažováno o žádném algoritmu z této kategorie. Druhou kategorii tvoří jednofázové algoritmy. Jak název napovídá, jednofázové algoritmy provádí detekci v jediném kroku. Obvykle umístí na obrázek mřížku a každá její buňka předpovídá několik bounding boxů. Jednofázové algoritmy umožňují detekci v reálném čase a zachovávají si srovnatelnou úspěšnost [61]. Nejvýznamnější algoritmy z této kategorie jsou SSD (Single Shot Detection) a YOLO (You Only Look Once). Oba algoritmy používají mřížku a konvoluční neuronové sítě pro regresi bounding boxů a jejich tříd. Realtime modely používající novější verze YOLO algoritmu dosahují lepších výsledků v rychlosti a podobné nebo vyšší v přesnosti oproti srovnatelným SSD modelům [[61], [63], [64]].

Originální YOLO algoritmus [65] rozdělí obrázek na mřížku o velikosti $S \times S$. Buňka mřížky je zodpovědná za detekci objektu, který má střed uvnitř buňky. Každá buňka předpovídá B bounding boxů a skóre důvěry (confidence score) vyjadřující úroveň přesvědčí o výskytu objektu v bounding boxu. Každý bounding box je definovaný pomocí souřadnic středu x a y , šířky w , délky h a důvěry (confidence). Souřadnice středu jsou relativní k hranicím buňky. Výška a šířka jsou relativní k obrázku. Každá buňka také předpovídá pravděpodobnost třídy označovanou C . Buňka vždy předpovídá pouze objekty z jedné třídy. Pro odstranění nadbytečných bounding boxů se obvykle používá NMS (Non Max Suppression) [62]. NMS odstraňuje všechny bounding boxy, které mají větší překrytí s jiným bounding boxem, než je práh a mají menší skóre důvěry. Úroveň překrytí vyjadřuje metrika IoU (Intersection over Union – průnik nad sjednocením) definovaná jako poměr obsahu průniku dvou bounding boxů a celkového obsahu sjednocení obou bounding boxů.

YOLOv2 [66] se zaměřilo na zmenšení lokalizačních chyb. Přidává vylepšení ve formě dávkové normalizace, dalšího tréninku na obrázcích s větším rozlišením, odstranění plně propojených vrstev a používání anchor boxů. Anchor boxy jsou výchozí bounding boxy různých tvarů a poměrů stran [67]. V YOLOv2 se místo rozměrů bounding boxů předpovídají posuny od rozměrů anchor boxů. YOLOv3 [68] přidává další menší vylepšení jako změna sítě pro extrakci příznaků, nezávislé logistické klasifikátory pro určení třídy a odhadování bounding boxů ve třech měřítkách. YOLOv3 je poslední verze od původních autorů. Další verze na sebe nijak navenavazují a jedná se obvykle o nezávislé projekty odvozené nebo různým způsobem navazující na původní 3 verze YOLO. Vyšší verze tedy neznamená nutně vylepšení. V době výběru byly dostupné například verze YOLOv4, YOLOv5 a YOLOR. Během vývoje vznikly ještě YOLOv6 a YOLOv7. Verze YOLOv4 je dle jednoho z původních autorů považována za kanonickou verzi [70]. Všechny projekty obsahují odlišné velikosti modelů, kde větší modely jsou přesnější a mají menší rychlejší.

Vzhledem k požadavkům a náročnosti detekce objektů byla při výběru výrazně prioritizována rychlost inference, která je pro systém kritická. Z toho důvodu bylo vždy hleděno na ty nejmenší modely. Samozřejmě model musí nabízet i dostatečnou úspěšnost. Porovnání úspěšnosti modelů je značně obtížné. Výsledky poskytnuté autory nejsou průkazné a různé benchmarky mají pro stejné modely odlišné výsledky. Důvodem je velké množství neznámých detailů a faktorů, které výsledky ovlivňují. Tyto výsledky proto spíše naznačují schopnosti modelu. Při výběru byl brán ohled i na stav dostupné implementace, její snadnou integraci do systému a jednoduchost změny velikosti modelu.

Vybrány byly nakonec modely z rodiny YOLOv5. Jedná se o projekt autora PyTorch implementace YOLOv3. Hlavním důvodem výběru byla varianta YOLOv5n, která z uvažovaných možností nabízí nejmenší a nejrychlejší model. Ostatní možnosti nenabízeli přijatelnou rychlost. Během vývoje vyšlo YOLOv7, které slibovalo výrazně vyšší rychlost i přesnost než YOLOv5 [71]. Při vlastním testu ovšem YOLOv7 nedosahovalo očekávaných rychlostí a bylo pomalejší než YOLOv5n. Benchmark Amitabha Banerjee také ukazoval podobnou rychlost i přesnost YOLOv5 (ve variantě s) a YOLOv7 [72].



Graf 2 – Výsledky vlastního testování rychlosti modelů

Velkými výhodami YOLOv5 je kvalitní implementace s velkým množstvím nástrojů, snadná integrace pomocí PyTorch Hubu a snadná změna velikosti modelu změnou jediného řetězce. Žádnou z těchto vlastností ostatní varianty nenabízí. Při testech detekoval model osoby poměrně úspěšně za většiny okolností. Horší výsledky se objevovali při nízké kvalitě obrazu. Další problém představovaly davy lidí. Ve výsledku byla úspěšnost shledána dostatečnou. Výchozí hodnota pro práh skóre důvěry byla empiricky nastavena na 0,4.

System nevyžaduje velkou přesnost rozměrů bounding boxů. Tělo osoby musí být zachycené celé a potencionální nadbytečné pozadí není podstatné. Proto jsou rozměry bounding boxů rozšířené z každé strany 25 procenty hodnoty daného rozměru ve výchozím nastavení.

3.5.2 Sledování objektů

Sledování, stejně jako další části systému, vyžaduje velmi rychlou, ale spolehlivou metodu. Tu představuje algoritmus SORT (Simple Online and Realtime Tracking) [74]. SORT nabízí sledování několika objektů s vysokou úspěšností v minimálním čase. Jedná se o jeden z nejvýznamnějších algoritmů pro sledování objektů. K sledování používá rozměry a polohu bounding boxů. Z toho důvodu je náchylný na záměnu objektů v situacích, kdy se dva objekty překrývají nebo pohybují ve velké blízkosti. Tento problém řeší algoritmus DeepSORT, který původní SORT rozšiřuje a přidává k němu vizuální informaci získávanou konvoluční

neuronovou sítí [75]. Přesto byl pro systém vybrán SORT, protože je z důvodu absence neuronové sítě znatelně rychlejší. Pokud záměna nastane v průběhu identifikace, tak naruší celý proces, ale mimo něj nemá podstatný význam. Systém využívá vlastní implementaci SORT algoritmu.

Základním principem SORTu je asociace předpovězených bounding boxů se skutečnými. Vztah poloh objektu mezi snímky modeluje lineární model s konstantní rychlostí (ve významu změny veličiny - velocity) nezávislí na pohybu kamery a ostatních objektů. [74]

Model je definován jako:

$$\mathbf{x} = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]^T$$

Kde u je horizontální souřadnice středu, v vertikální souřadnice středu, s obsah a r poměr stran. T označuje časový údaj vyjádřený snímky. Ostatní proměnné označují rychlost změny jednotlivých veličin v intervalu mezi měřeními. Rychlost je v tomto intervalu považována za konstantní. Poměr stran r je konstanta. Odhady jednotlivých atributů vytváří Kálmánův filtr. Kálmánův filtr je algoritmus pro odhad stavu systému z dat, která mohou obsahovat nepřesnosti měření nebo šum. Jedná se o významný algoritmus používaný k sledování objektů, odhadu změn veličin, navigaci a dalším činnostem. Odhadnuté atributy jsou použity pro přiřazení detekce a stopy. Po úspěšném přiřazení aktualizují hodnoty detekovaného bounding boxu parametry Kálmánův filtru, který odhadoval polohu sledovaného objektu. Kálmánův filtr dokáže vyhodnocovat váhu současného odhadu a naměřených hodnot. Neuchovává si minulé odhady a veškeré informace používané pro odhady jsou zahrnuty v jeho stavu. V důsledku toho není náročný na paměť. Ve vlastní implementaci byl použit Kálmánův filtr z knihovny FilterPy. Parametry filtru byly určeny empirickými experimenty z vlastními videozáznamy chůze a z MOT20 benchmarkem. [75], [76]

V ideální situaci, kdy je objekt detekován v každém okamžiku pohybu, odhad probíhá pouze ze snímku na snímek a poté je stav modelu aktualizovaný skutečnými naměřenými hodnotami. Pokud na snímku objekt není detekován, Kálmánův filtr musí odhadovat delší sekvenci hodnot a začíná se projevovat nedokonalost modelu, který nedokáže zachytit skutečnou dynamiku pohybu a nemůže reagovat na změny v pohybu [74]. Pro každý nový objekt je vytvořena stopa, která obsahuje id, Kálmánův filtr a počet po sobě jdoucích snímků, na kterých nebyl objekt

nalezen. Pokud není objekt nalezen na počtu snímků větším než je hodnota prahu, stopa je odstraněna. Práh má v systému výchozí hodnotu 5.

Asociaci předpovězených bounding boxů stop a skutečných provádí Maďarský algoritmus. Pro vyjádření podobnosti bounding boxů využívá metriku IoU. Maďarský algoritmus [73] hledá optimální přiřazení do dvojic tak, aby součet cen všech dvojic byl nejmenší možný. Před aplikací algoritmu musí být hodnoty IoU všech kombinací odhadnutých a skutečných bounding boxů umístěny do matice $M \times M$. Kde M je maximum z počtu a detekcí a stop. Hodnoty jsou do matice umístěny tak, že je matice naplněna hodnotami 1 a odpovídající IoU jsou od nich odečteny. Tím se hodnoty převrátí. Důvodem k tomu je skutečnost, že Maďarský algoritmus hledá přiřazení s nejnižší možnou celkovou cenou.

Přiřazování probíhá následujícím způsobem [73]:

1. Odečtení minima ze všech řádků matice.
2. Odečtení minima ze všech sloupců matice.
3. Překrytí nul minimálním počtem horizontálních a vertikálních čar přes celý řádek/sloupec.
4. Pokud je počet čar rovný M , algoritmus končí a ideální přiřazení lze poskládat z párů odpovídajícím umístění nul.
5. Z hodnot nezakrytých žádnou čarou se vybere minimum. Odečte se od všech nezakrytých elementů a přičte ke všem nulám zakrytým více čarami.
6. Algoritmus se vrací k 3. kroku.

V implementaci je hledání optimálních čar řešeno hladovým algoritmem, který vždy hledá čaru zakrývající největší počet nul, dokud nejsou zakryty všechny nuly. Zakrývání elementů čarami reprezentuje matice ve tvaru $M \times M$, kde hodnota elementu odpovídá počtu zakrytí. Finální řešení vzniká výběrem nul, tak aby v každém sloupci a řádku byla vybrána pouze jedna nula. Implementace procesu spočívá v postupném výběru nulových elementů unikátních v sloupci nebo řádku a maskování řádků a sloupců s vybraným párem v dalších iteracích. Pokud takový element neexistuje, tak je vybrán náhodný nulový element. V každé iteraci se výběr provede pro všechny sloupce nebo pro všechny řádky a na konci se aktualizuje seznam

nulových elementů. Vyberou-li se sloupce nebo řádky určuje počet vhodných případů pro výběr.

3.5.3 Vytváření siluet

Vytváření siluet je operace, která je zásadní pro celý proces. Silueta je v rámci systému definována jako obrázek, kde pixely lidské postavy mají hodnotu 255 a pixely pozadí hodnotu 0. Musí být dostatečně přesné, nesmí obsahovat výrazný šum a systém je musí být schopný vytvářet rychle.

Tradičně se vytvářeli pomocí procesu odstranění pozadí (background subtraction), při kterém dochází k modelování pozadí. Obvykle se sledují změny vůči modelu pozadí a veškeré pohybující se objekty jsou považovány za popředí. Nejznámější metody odstranění pozadí jsou MoG (Mixture of Gaussians), KNN a jednoduchý rozdíl snímků. Pokročilejší metodu představuje například SuBSENSE [69]. Metody byly testovány s neuspokojivými výsledky. Odstraňování pozadí je velmi závislé na prostředí, má náchylnost k šumu, nízkou odolnost vůči dešti a větru, vyžaduje vhodné nasvícení a dochází k problémům jako zahrnování osob do pozadí.

Proto bylo nutné vybrat odlišný přístup. Vytváření siluet lze považovat za podmnožinu sémantické segmentace. Sémantická segmentace je typ úlohy, kdy se jednotlivé pixely obrázku přiřazují k třídám. Pro systém byla zvolena hluboká neuronová síť pro sémantickou segmentaci Fast-SCNN [78]. Fast-SCNN je efektivní plně konvoluční neuronová síť s nízkou kapacitou navržená pro segmentaci v reálném čase. Autoři ji testovali na městské krajině, kde mimo jiné ukázala dobrou přesnost segmentace osob na ulici.

Systém používá vlastní implementaci sítě, která byla upravena pro potřeby systému. Na výstupu byla poslední vrstva vyměněna za konvoluční vrstvu s jedním filtrem a aktivační funkcí sigmoid, aby síť prováděla binární klasifikaci každého pixelu. Rozměr vstupního obrázku byl za účelem zvýšení rychlosti zmenšen na rozlišení 256×128. Poměr stran byl zvolen s ohledem na poměr stran očekávaného vstupu. Obrázky si při převodu do rozměrů vstupu zachovávají poměr stran a potřebný prostor se doplňuje nulami. Vzhledem k zmenšení vstupu byl u jedné z prvních konvolučních vrstev snížen krok, aby si mapy příznaků v nejhlubších vrstvách zachovali dostatečně velké rozměry. Experimenty s přidáním další větve nepřinesly výsledky.

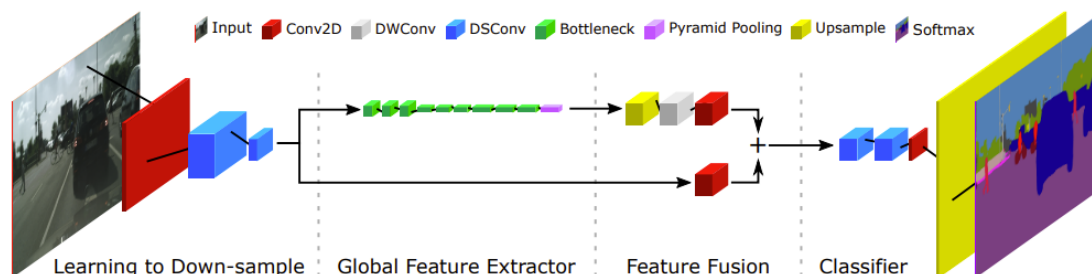
Fast-SCNN kombinuje architekturu s dvěma větvemi a kodér-dekodér. Architektura s dvěma větvemi pracuje s předpokladem, že komplexní korelace mezi třídami objektů (tzv. globální kontext) se síť učí ve vrstvách, kde jednotlivá konvoluční jádra zpracovávají větší prostor. Naopak ve vrstvách zpracovávajících menší prostor se učí detaily jako jsou hranice objektů. To znamená, že hluboká větev s menším rozlišením zachycuje globální kontext a mělká větev s plným rozlišením prostorové detaily. Kodér-dekodér nejprve kóduje vstup do kompresované reprezentace a tu následně mapuje na požadovaný výstup [39]. Autoenkodér je specifickým případem kodéru-dekodéru. V kontextu segmentace obrazu kodér-dekodér často používá tzv. skip connection. Skip connection označuje zapojení výstupu vrstvy neuronů tak, že přeskakuje jiné vrstvy. Řeší problém mizení gradientu a prokazuje schopnost obnovovat detaily v hlouběji umístěných vrstvách [79]. [78]

Fast-SCNN využívá hloubkově oddělitelnou konvoluční vrstvy, inverted residual bottleneck bloky a modul pyramidového poolingů. Hloubkově oddělitelná konvoluční vrstva funguje jako odlehčená a zrychlená konvoluční vrstva. Nejprve provede konvoluci na každém kanále nezávisle a následně kanály sloučí pomocí konvoluce s jádrem 1×1 [28]. Tím dosáhne výrazného snížení počtu parametrů. Inverted residual bottleneck blok představený v síti MobilNetV2 [77] je blok se skip connection, který se dokáže efektivně učit a řešení problému mizení gradientu.

Modul pyramidového poolingů [80] slouží k získání informací o globálním kontextu. Provádí pooling s funkcí průměr (average pooling) na několika úrovních s rozdílnými rozměry jádra, které rozděluje vstup na oblasti. Každá úroveň vyprodukuje výstup s určitým počtem oblastí. Všechny výstupy se převedou na původní rozměry a sloučí se vstupem tak, aby celkový počet filtrů vzniklých map příznaků byl rovný počtu filtrů vstupu. Počet oblastí na jednotlivých úrovních ve vlastní implementaci je 1×1 , 2×2 , 4×4 a 8×8 .

Fast-SCNN sdílí první vrstvy pro obě větve, protože zachycují nízko-úrovňové příznaky jako jsou hrany. Snižuje tím výpočetní náročnost sítě. Celá síť je rozdělena do modulů. Extrakci nízko-úrovňových (lokálních) příznaků obstarává *Learning to Down-sample* modul složený z jedné konvoluční a dvou hloubkově oddělitelných konvolučních vrstev s krokem 2. Následuje *Global Feature Extractor* modul pro získání globálních příznaků. Ten obsahuje bottleneck bloky zakončené modulem pyramidového poolingů. *Feature Fusion Module* pak slouží k sloučení lokálních příznaků a globálních příznaků. Sloučení probíhá sjednocením rozměrů pomocí zvětšení globální mapy příznaků, sjednocením počtu filtrů

a následným součtem obou map příznaků po složkách. Posledním modulem je *Classifier*, který provádí závěrečnou klasifikaci. Obsahuje dvě hloubkově oddělitelné vrstvy a výstupní konvoluční vrstvu s aktivační funkcí sigmoid v případě vlastní implementace. [78]



Obrázek 9: Architektura Fast-SCNN sítě [78]

Pro učení sítě byla použita kombinace datasetů COCO, Segmentation Full Body TikTok Dancing Dataset a Penn-Fudan Dataset for Pedestrian Detection and Segmentation. COCO dataset byl zredukován na obrázky obsahující lidi. Data byla převedena tak, aby anotace byly ve formátu binárních masek. Pro trénování byly jednotlivé osoby vyříznuty a uloženy do samostatných souborů. Vyříznutí osob z obrázků simuluje bounding boxy náhodným zvětšením hranic výřezu. Data byla doplněna prázdnými výřezy pozadí. Validací množina je tvořena především COCO validačním datasetem doplněným daty z dříve uvedených datasetů.

První testování odhalilo problémy na určitých pozadích. Konkrétními příklady těchto pozadí jsou některé stromy a jejich větve, dřevěné klády, komplikovanější interiéry a obecně pozadí s komplikovanou texturou. Problém byl vyřešen generováním nových obrázků umístěním osob vyříznutých podle masky na různá pozadí. Umístění osob probíhá výměnou pixelů pozadí v místech určených maskou. Osoby mají náhodnou velikost a náhodné umístění v pozadí. Pro lepší zakomponování osob do pozadí byly masky zpracovány morfologickou operací eroze, která zúží jednotlivé části siluety, a Gaussovským rozostřením. Výsledná silueta má hranice změkčené nižšími hodnotami pixelů. Barva pixelu osoby ve vytvořeném obrázku je určena váženým součtem po složkách osoby a pozadí, kde váhu (v procentech) udává hodnota pixelu siluety. Dalšího zdokonalení výsledného obrázku bylo dosaženo přenosem barev z pozadí na přidávanou osobu. K tomu byla použita metoda, kterou vytvořili E. Reinhart et al. [81].

Kroky metody přenosu barev:

- Převedení obou obrázků do barevného prostoru Lab , kde osa L vyjadřuje světelnost, osa a jsou barvy zelená až červená, b je osa s barvami modrá až žlutá.
- Odečtení průměru každého kanálu obrázku obsahujícího osobu z jeho odpovídajících kanálů.
- Vynásobení výsledku poměrem směrodatných odchylek pozadí a výsledku.
- Přičtení průměrů kanálů pozadí ke kanálům výsledku.
- Převedení výsledného obrázku zpět do RGB barevného prostoru.

Bylo zjištěno, že převod barev je příliš výrazný. Proto byl obrázek s převedenými barvami zkombinován s původním obrázkem se stejnými váhami. Pro generování obrázků byl vytvořen vlastní dataset, který obsahuje 131 obrázků rozmanitých prostředí. Osoby byly z datasetů TickTok Dancing a Penn-Fudan, které obsahují celá těla osob a jsou nejbližší očekávaným vstupům. Zvýšení význam právě těchto dat je žádoucí. Výsledný počet obrázků je 42 798.



Obrázek 10: Vlevo původní obrázek [52], vpravo obrázek po změně pozadí

Načítání dat provádí vlastní data generátor. Načítá obrázky po dávkách a provádí augmentaci. V každé epoše s výjimkou první probíhá výrazná augmentace. Augmentace označuje proces vytváření odlišných dat úpravou těch dostupných. Typickými augmentacemi jsou otočení, změna jasu nebo změna kontrastu. Nastavení augmentace bylo provedeno empiricky. Do řešení ji dodává knihovna Albumentations.

```

A.RandomSizedCrop(p=0.3, width=dimensions[1], height=dimensions[0],
                  min_max_height=[int(dimensions[0] / 2), dimensions[0]], w2h_ratio=0.5),
A.HorizontalFlip(p=0.5),
A.Rotate(p=0.2, limit=[-45, 45], border_mode=cv2.BORDER_CONSTANT),
A.GaussianBlur(p=0.05),
A.MedianBlur(p=0.1),
A.GaussNoise(p=0.12, var_limit=[10, 60]),
A.MotionBlur(p=0.15, blur_limit=[3, 7]),
A.RandomBrightnessContrast(p=0.4),
A.RandomContrast(p=0.08, limit=[-0.1, -0.3]),
A.Downscale(p=0.1, scale_max=0.8, scale_min=0.5),
A.RGBShift(p=0.2),
A.ToGray(p=0.02)

```

Obrázek 11: Seznam augmentací, parametr p označuje pravděpodobnost výskytu

Na rozdíl od původního návrhu nebyl použit před poslední vrstvou dropout. Dropout je regularizační technika, která způsobuje ignorování části výstupů vrstvy a snižuje tím přeučení. Přeučení označuje problém, kdy se model učí příklady příliš detailně a není schopný generalizovat. Bylo zjištěno, že jediný efekt dropout ve Fast-SCNN síti spočívá v odlišném poměru metrik recall a precision, který je bez dropoutu vyrovnanější, a drobném snížení přesnosti. Recall je množství skutečně pozitivních labelů, které se podařilo klasifikovat správně. Precision označuje množství skutečně pozitivních labelů mezi pozitivně klasifikovanými labelem. Obdobně jako v původním návrhu sítě byl použit optimalizátor SGD a polynomiální pokles parametru learning rate. Learning rate je parametr optimalizace modelu strojového učení, který označuje krok k minimu chybové funkce. Parametry navržené autory se osvědčily a byly doplněny o konečný learning rate s hodnotou 0.0001. Trénink s 42 798 obrázky probíhal v 150 epochách a počet kroků poklesu byl nastaven na 40 000. Validační množina obsahovala 2 673 obrázků.

Learning rate s polynomiálním poklesem v kroku s [82]:

$$s = \min(s, d)$$

$$learning\ rate = (i - e) * \left(1 - \frac{s}{d}\right)^{0.9} + e$$

- i – počáteční learning rate (0.045)
- e – konečný learning rate (0.0001)
- d – počet kroků poklesu (40 000)

Přínos poklesu parametru learning rate spočívá v zamezení uváznutí v nežádoucích lokálních minimech a zrychlení tréninku v počátečních fázích tréninku s doladěním v pozdějších fázích. Jako chybová funkce slouží binární křížová entropie (binary cross-entropy).

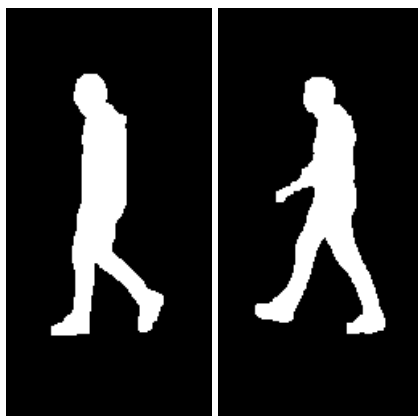


Obrázek 12: Ukázka výstupy Fast-SCNN

Síť byla testována na datasetu Persons Labeled. Dosáhla přesnosti 96,71 %, precision 90,74 %, recall 88,61 %, F1 skóre 89,66 % a IoU 88,71 %. F1 skóre je metrika kombinující recall a precision vhodná pro nevybalancované třídy. Inference modelu v prostředí ONNX v dávce velikosti 1 dosahuje trvání 18,48 ms. V OpenVINO prostředí trvá inference 15 ms.

3.5.4 Detekce cyklu chůze

Pro řešení detekce chůze byla navržena vlastní metoda. Metoda využívá periodického střídání poloh nohou během chůze. Poloha nohou udává fázi kroku. Pro systém jsou významné dvě fáze, které jsou nazvány *Mid* a *Heel strike*. Jejich názvy jsou inspirovány fázemi krokové cyklu.



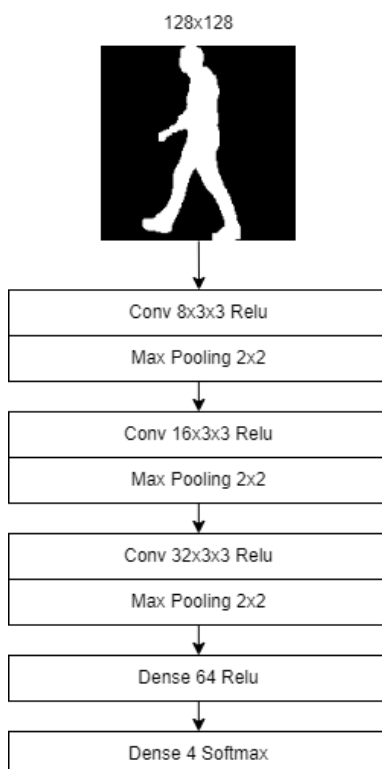
Obrázek 13: Vlevo fáze Mid, vpravo fáze Heel strike

Tyto fáze kroku byly zvoleny, protože jsou snadno odlišitelné vzájemně i od zbylých fází. Další fáze jsou označeny jako *Other*. Mimo nich existuje ještě fáze *Unclassified*, která označuje neznámé aktivity.

Během chůze se fáze střídají v pořadí *Heel strike*, *Other*, *Mid*, *Other*. Cyklus chůze může začít fází *Mid* nebo *Heel strike*. Sekvence obrázků je považována za cyklus, pokud probíhá ve správném pořadí a počáteční fáze se v sekvenci objeví třikrát. Detekce stejné fáze za sebou se ignorují. Cyklus končí, když se objeví poslední detekce počáteční fáze cyklu. Cyklus je zrušen, pokud se čtyřkrát za sebou objeví fáze *Unclassified*.

Metoda je implementována jako stavový automat. Snímky siluet jsou postupně vkládány do automatu a následně probíhá klasifikace fáze kroku, podle které se mění stav. Fáze *Other* nikdy nemění stav, aby byla metoda odolnější proti špatným klasifikacím. Snímky se ukládají, dokud nedojde k resetu.

Pro klasifikaci fází kroku byla navržena konvoluční neuronová síť. Na návrhu se projevil požadavek na rychlost. Vrstvy obsahují poměrně malý počet neuronů.



Obrázek 14: Síť pro klasifikaci fáze kroku

Pro trénování sítě byl vytvořen dataset pro klasifikaci fáze chůze z datasetů CASIA A, CASIA B a CASIA C. Do jednotlivých kategorií byly siluety zařazeny tak, aby kategorie obsahovaly

všechny zaměnitelné snímky fáze. Do kategorie *Unknown* byly zařazeny přední, zadní pohled a některé obrázky z datasetů pro vytváření siluet. Při tvorbě datasetu byly obrázky zařazovány do trénovací a validační množiny. Trénovací množina se skládala z 4391 obrázků rovnoměrně rozdělených do příslušných kategorií. Validací množina obsahovala 980 obrázků.

Trénink probíhal v 30 epochách a s dávkou velikosti 12. Po tréninku se ukládají váhy z epochy s nejlepšími výsledky. Learning rate byl empiricky nastaven na 0,01. Kategorická křížová entropie byla použita jako chybová funkce. Pro lepší generalizaci byla empirickým zkoumáním vybrána l2 regularizace s hodnotou činitele 0,001. Regularizace l2 snižuje váhy sítě a omezuje tím přeučení.

Testování proběhlo na validační množině, která obsahuje reprezentativní a rozmanitý vzorek příkladů.

ODHADNUTÉ/SKUTEČNÉ	Heel strike	Mid	Other	Unknown
Heel strike	235	1	10	2
Mid	1	247	2	5
Other	5	3	236	10
Unknown	0	0	3	220

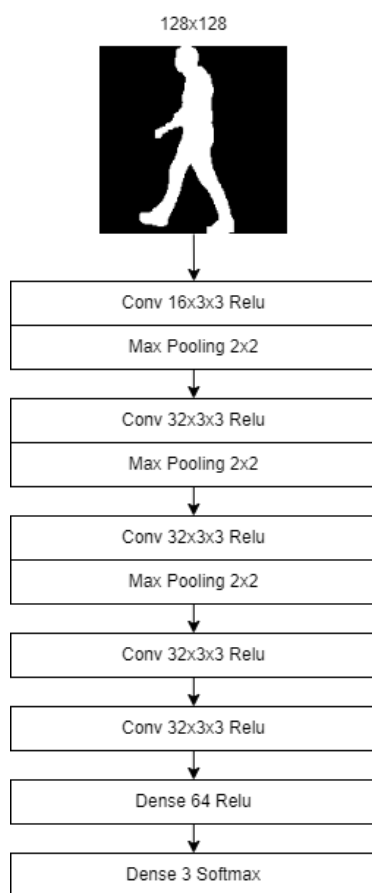
Tabulka 1: Matice záměn klasifikace fáze kroku

Síť udělala 42 chyb během 938 klasifikací. Dosáhla tak přesnosti 95,71 %. Významné je především, že pouze dvakrát zaměnila kategorie *Heel strike* a *Mid*, jejichž odlišení je nejpodstatnější. Inference modelu v prostředí ONNX v dávce velikosti 1 dosahuje trvání 0,36 ms.

3.5.5 Detekce směru

Sjednocení směru chůze přináší výrazné usnadnění pro samotné rozpoznávání. Pro zdokonalení spolehlivosti má chůze klasifikovaný směr, pokud se objeví v 5 snímcích za sebou. Klasifikaci směru provádí navržená konvoluční síť. Stejně jako u ostatních částí

systému je cílem dostatečná rychlost. Kategorie pro klasifikaci jsou směr vlevo, vpravo a dopředu/dozadu.



Obrázek 15: Síť pro klasifikaci směru

Pro trénovací množinu byly náhodně vybrány snímky z datasetu CASIA B. Pro směr vlevo i vpravo byly použity stejné obrázky převrácené. Stejným způsobem byly vybrány obrázky pro validační množinu z datasetu CASIA A a pro testovací množinu z datasetu CASIA C. Testovací množina byla doplněna obrázky předního a zadního pohledu z datasetu OUMVLP.

Trénink probíhal 20 epoch s velikostí dávky 8 a learning rate 0.001. Jako chybová funkce byla použita kategoričká křížová entropie. Totožně s neuronovou sítí pro klasifikaci fází chůze byla použita l2 regularizace se stejným činitelem.

Výsledky testování:

ODHADNUTÉ/SKUTEČNÉ	Přední/zadní	Levý směr	Pravý směr
Přední/zadní	450	6	11
Levý směr	28	674	8
Pravý směr	122	14	675

Tabulka 2: Matice záměn klasifikace směru chůze na validačním datasetu

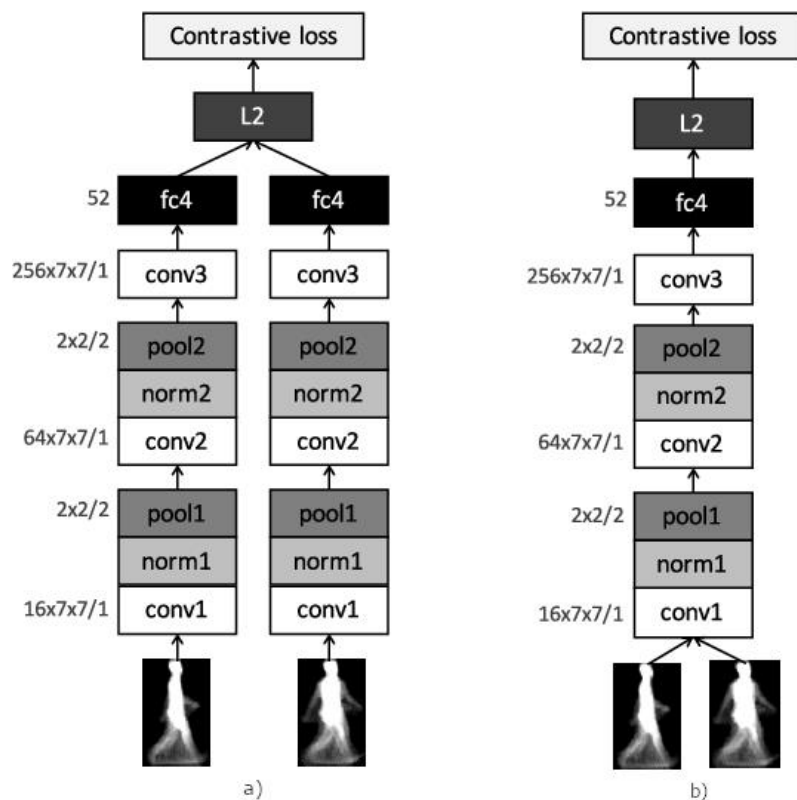
ODHADNUTÉ/SKUTEČNÉ	Přední/zadní	Levý směr	Pravý směr
Přední/zadní	1184	11	44
Levý směr	64	1188	196
Pravý směr	133	263	1222

Tabulka 3: Matice záměn klasifikace směru chůze na testovacím datasetu

Síť dosáhla přesnosti 90,5 % na validačním datasetu a 83,5 % na testovacím datasetu. Při hodnocení výsledků je nutné přihlédnout k nižší kvalitě některých siluet v obou testovaných datasetech. Inference modelu v prostředí ONNX v dávce velikosti 1 dosahuje trvání 0,87 ms.

3.5.6 Rozpoznání osob

Rozpoznávání osob podle chůzí spočívá v určení odlišnost (podobnosti) dvou chůzí, podle které lze určit, zda chůze náleží stejné osobě a případně identifikovat osobu. Systém používá vlastní implementaci neuronové sítě, kterou navrhli N. Takemura et al. [35]. Řešení bylo vybráno z důvodu vysoké úspěšnosti, přijatelné rychlosti a využití formátu GEI pro vstupy. Autoři navrhli několik sítí a jejich kombinace. Všechny sítě vychází ze sítí *diff* a *2in*, které jako chybovou funkci využívají contrastive loss. Síť *diff* je konvoluční síť, která má na vstupu rozdíl porovnávaných GEI a L2 normu (Eukleidovská norma) na výstupu. L2 norma udává Euklidovskou vzdálenost vektoru od počátku. Síť *2in* představuje siamskou síť, jejíž dva vstupy jsou porovnávané GEI. Výstupem je Euklidovská vzdálenost výstupních map příznaků obou podsítí.



Obrázek 16: a) 2in b) diff [35]

V důsledku toho, že v *diff* síti porovnání probíhá v první vrstvě na nízké úrovni, dosahuje lepší výsledků, pokud je rozdíl v úhlech porovnávaných GEI menší. Příznaky na nízké úrovni mají větší citlivost na některé odlišnosti. Naopak v *2in* síti jsou porovnávány vysoko-úrovňové příznaky v poslední fázi, proto je robustnější a dosahuje lepších výsledků u větších rozdílů v úhlech. [35]

Další sítě jsou varianty používající triplet loss a kombinace obou sítí v obou variantách chybové funkce. Kombinování sítí vzniká zprůměrováním jejich výsledných hodnot odlišnosti. Sítě s triplet loss chybovou funkcí mají jeden vstup navíc. Srovnatelné sítě používající contrastive loss dosahují lepších výsledků při verifikaci a ty využívající triplet loss mají lepší výsledky v identifikačních scénářích. Systém by měl mít dobré výsledky pro identifikaci i verifikaci. Za mírně důležitější je považována verifikace. Prioritizace verifikace povede k nižšímu počtu FP chyb. Kombinace sítí dosahují nejlepších výsledků. Zlepšení se týká všech scénářů použití i rozdílů v úhlech. [35]

Vybrána byla síť *2in + diff* (v rámci systému nazývána *diff2in*), která kombinuje sítě uvedené v názvu a používá chybovou funkci contrastive loss. Srovnávání provedená autory ukazují, že má nejlepší výsledky pro verifikaci a pro identifikaci překonává samostatné sítě s contrastive loss. Výsledky identifikace ve srovnání se sítěmi používajícími triplet loss se liší v závislosti

na datasetu a rozdílu úhlů. Za znatelně lepší lze v identifikačním scénáři považovat kombinovanou síť s triplet loss nazývanou *3in + 2diff*. Výsledky zvolené sítě jsou ovšem v tomto směru považovány za dostatečné. Zvolená síť mi oproti síti *3in + 2diff* výhodu menších rozměrů a používání dvou vstupních obrázků.

Contrastive loss použitá pro trénování [35]:

$$L = \frac{1}{2M} \sum_{m=1}^M \delta_{y1_m y2_m} d_m^2 + (1 - \delta_{y1_m y2_m}) * \max(\text{margin} - d_m^2, 0)$$

- M – počet párů
- $y1_m, y2_m$ – ID subjektů m -tých vstupních GEI
- δ – Kroneckerovo delta, $\delta = 1$ v případě, že jsou vstupní GEI od stejného subjektu a $\delta = 0$ u odlišných subjektů
- margin – autoři empiricky nastavili na hodnotu 3
- d_m – Euklidovská vzdálenost/norma m -tých vstupních GEI

Pro trénování byly použity datasety OUMVLP a CASIA B. Oba datasety byly rozděleny na trénovací a validační množinu. Validací množina obsahuje 823 subjektů z datasetu OUMVLP a 20 z datasetu CASIA B. Místo GEI poskytnutých v datasetech byly GEI vytvořeny pomocí metod popsanych v předešlých kapitolách. V důsledku toho GEI blíže odpovídaly očekávaným vstupům. Byly použity pouze siluety pořízené v úhlech mezi 54 a 126 stupni. Před tréninkem jsou vždy vytvořeny páry. Způsob párování je převzatý z článku o modulu PSTN [83] pro řešení rozdílů v úhlech, jejíž autoři použili jako benchmark síť *diff* a *2in*. Každý GEI subjektu je párován se všemi jeho ostatními GEI a 9 GEI odlišného náhodné subjektu v náhodném úhlu. Dataset OUMVLP obsahuje výrazně větší množství subjektů, ale s menším množstvím vzorků od jednotlivých subjektů a vždy ve stejném oblečení. CASIA B nabízí výrazně větší rozmanitost. Subjekty byly zachyceny v různém oblečení a s batohy nebo taškami. Pro maximální využití vlastností obou datasetů byl model nejprve natrénován na datasetu OUMVLP a následně doladěn na datasetu CASIA B. Tento způsob tréninku poskytoval nejlepší celkové výsledky. Nejprve byla síť trénována ve 2 epochách na OUMVLP datasetu. V první epoše s learning rate 0,01 a v druhé 0,001. Následuje 1 epocha

ladění s datasetem CASIA B a learning rate 0,01. Po ladění dosahuje síť o 7,24 % nižší úspěšnost na validační množině OUMVLP, ale na validačním datasetu CASIA B má přesnost lepší o 17,24 %.

Používány byly dávky o velikosti 200 z důvodu omezeného množství paměti grafické karty. Síť byla trénována jako celek.

Testování bylo provedeno na datasetech CASIA A a CASIA C. Z datasetu CASIA A byly použity úhly 0° (odpovídá 90° v CASIA B) a 45°. Při interpretaci je nutné zohlednit, že kvalita siluet v testovacích datasetech je nižší a síť nebyla trénována na úhel 45°. Mimo to neobsahovaly tréninkové datasety rozdílné rychlosti chůze na rozdíl od datasetu CASIA C. Inference modelu v prostředí ONNX v dávce velikosti 1 dosahuje trvání 67,24 ms.

Výsledky finálního modelu:

	OUMVL P	OUMVL P 75°- 105°	OUMVL P 90°	CASIA B	CASIA B 72°- 108°	CASIA B 90°	CASIA C	CASIA A	CASIA A 90°
Počet párů	775 600	378 400	87 600	330 400	103 800	8 400	68 000	8 800	1 400
Chyby	70 628	25 384	4 131	4 975	24 980	302	12 516	24 79	322
EER	0,067	0,091	0,047	0,048	0,076	0,036	0,184	0,281	0,230
Přesnost (%)	90,89	93,29	95,28	92,44	95,2	96,40	81,59	71,82	77

Tabulka 4: Výsledky vlastní implementace diff2in na validačních a testovacích datasetech s prahem, který dosahuje EER

Jako výchozí práh pro systém byl zvolen práh s poměrem 6 : 1 False Negative a False Positive chyb. Důvodem je větší negativní význam False Positive chyb. Hodnota prahu byla určena jako průměr hodnot prahů obou validačních datasetů s žadáním poměrem chyb.

Výsledky finálního modelu s výchozím prahem:

	OUMVLP	CASIA B
Počet párů	775 600	330 400
False Negative	76 270	28 539
False Positive	12 713	4 757
Chyby	88 983	33 296
Přesnost (%)	88,52	89,92
Práh	1,9749	1,3803

Tabulka 5: Výsledky vlastní implementace diff2in na validačních datasetech s prahem na úrovni, kde poměr FN a FP je 6 : 1

	CASIA C	CASIA A
Počet párů	68 000	8 800
False Negative	15 961	2 876
False Positive	932	232
Chyby	16 893	3 108
Přesnost (%)	75,15	64,68

Tabulka 6: Výsledky diff2in s prahem 1,6776 na testovacích datasetech

3.6 Testování

Kapitola se zabývá celkovým testováním systému. Testování jednotlivých částí je k nalezení v minulé kapitole.

System byl testován na vlastních videích, která obsahují 5 osob k rozpoznání. Doplnují je 3 videa neznámých subjektů. Celkový počet videí je 19. Osoby jsou zachyceny z boku v různých prostředí a v některých případech v odlišném oblečení. Testování probíhá v režimu způsobujícím, že po každém dokončení identifikace následuje nová identifikace. Testována byla zároveň identifikace i verifikace se všemi subjekty. Veškeré testování proběhlo na

testovacím zařízením a ve 30 snímcích za sekundu. Testovací zařízení zachytilo 40 detekcí oproti 80 detekcím běhu ve 30 snímcích za sekundu.

Verifikace:

ODHADNUTÉ/SKUTEČNÉ	Stejná osoba	Odlíšná osoba
Stejná osoba	28	16
Odlíšná osoba	5	151

Tabulka 7: Matice záměn verifikace osob systémem na testovacím zařízení

ODHADNUTÉ/SKUTEČNÉ	Stejná osoba	Odlíšná osoba
Stejná osoba	53	18
Odlíšná osoba	16	313

Tabulka 8: Matice záměn verifikace osob systémem ve 30 FPS

Identifikace:

Správná identifikace	Špatná identifikace	Neidentifikováno
28	1	5

Tabulka 9: Výsledky identifikace na testovacím zařízení. Neidentifikováno znamená, že byla osoba určena jako neznámá.

Správná identifikace	Špatná identifikace	Neidentifikováno
50	2	9

Tabulka 10: Výsledky identifikace ve 30 FPS. Neidentifikováno znamená, že byla osoba určena jako neznámá

Systém na testovacím zařízení zpracovával průměrně 5,83 snímků za sekundu. V praxi nebyl rozdíl v rychlosti mezi použitím výchozího ONNX prostředí a ONNX OpenVINO. Přesnost verifikace je 89,50 % na testovacím zařízení a 91,5 % ve 30 snímcích za sekundu. Recall na testovacím zařízení dosahuje 84,84 % a precision 63,63 %. Ve 30 snímcích za sekundu je recall 76,81 % a precision 74,65 %. Přesnost identifikace na testovacím zařízení dosahuje

82,35 % a ve 30 snímcích za sekundu 81,97 %. Rozdíly v oblečení neukazovaly během testování významný vliv. Výsledky ukazují, že nižší počet zpracovávaných snímků nemá znatelný vliv na úspěšnost systému. Rozdíl mezi testovanými variantami se ovšem významně projevuje v množství detekovaných cyklů chůze, kde v 30 snímcích za sekundu detekuje systém dvojnásobný počet detekcí. Počet detekcí, který systém zvládá lze považovat za použitelný, ovšem s přihlédnutím k podmínkám. Testování s nižším počtem snímků za sekundu ukázalo, že současný výkon systému je blízko hranici použitelnosti.

Součástí akce Noc vědců 2022 na Jihočeské Univerzitě byla ukázka systému a návštěvníci si mohli vyzkoušet, jestli je dokáže systém identifikovat. Návštěvník se nejprve prošel po určené ploše a byl importován do systému. Následně se prošel znovu a systém se ho pokusil identifikovat mezi ostatními návštěvníky. Skupina návštěvníků byla rozmanitá a obsahovala subjekty různého věku a pohlaví. Větší část tvořily děti. Z celkového počtu 54 subjektů dokázal systém přiřadit správnou identitu 39 a dosáhl tak úspěšnosti 72,2 %. U některých subjektů mohl výsledek negativně ovlivnit problém, který snižoval kvalitu importovaných GEI. Problém byl způsobený chybou, která vznikla úpravou specifickou pro akci. Problém se objevoval odhadem u nižších jednotek subjektů.

4 Závěr

4.1 Zhodnocení výsledků

Práci se podařilo splnit určené cíle. Rozpoznání osob pomocí chůze včetně současných metod bylo popsáno v teoretické části. Byl navrhnut a implementován automatický systém pro rozpoznávání osob pomocí chůze, který zvládá rozpoznání z bočního pohledu v reálném čase. Podařilo se prokázat, že je systém schopný běhu na běžných zařízeních. Testování ukázalo potencionál pro použití v prostředích a situacích s tolerancí pro nepřesnosti. Běh na testovací zařízení určeném pro demonstraci byl porovnán s ideálním během ve 30 snímcích za sekundu. Bylo zjištěno, že testovací zařízení dosahuje srovnatelné přesnosti, ovšem s nižším počtem detekcí cyklu chůze. Systém byl představen a otestován na akci Noc vědců 2022 na Jihočeské Univerzitě, kde potvrdil schopnost rozpoznávat osoby s určitou spolehlivostí.

Balíček obsahující systém může sloužit jako knihovna nebo framework. Předpokládáné je použití systému v odlišných aplikacích ve formě ovládacího prvku. Výchozí funkcionalita slouží především jako demonstrace technologie. Bez rozšíření nebo úprav lze systém použít pro zaznamenávání docházky. Systém může být použitelný v inteligentní domech jako způsob spouštění akcí pomocí chůze, kde přesnost není kritická a nepřesnosti lze snadno a intuitivně napravit například hlasovým příkazem nebo příkazem na telefonu. Systém má stále významný prostor pro zlepšení a rozšíření značně omezených možností použití.

Kromě samotného systému obsahuje balíček nástroje pro zpracování dat, implementace modelů a skripty pro jejich snadné učení.

4.2 Rozšíření a vylepšení

Systém má stále velký prostor pro zdokonalení. Úspěšnost je použitelná, ovšem pro velkou část účelů nedostatečná. Zajímavé vylepšení by v tomto směru poskytovalo řešení rozdílů v úhlech. V současné chvíli se GEI vkládají do neuronové sítě pro rozpoznání v nezměněné formě. Rozpoznávání osob se vstupními GEI, kde se úhly vstupů liší, je obtížné a negativně ovlivňuje úspěšnost [35]. Negativní vliv rozdílů v úhlech je úměrný velikosti rozdílu. Zajímavé řešení nabízí modul PSTN, který vytvořili C. Xu at el. [83]. Ten řeší rozdíly v úhlech převedením obou vstupů do zobrazení, které se nachází mezi oběma úhly. Užitečným

rozšířením by bylo přidání více modelů specializovaných na různé podmínky a scénáře. Například přidání modelu $3in + 2diff$, který měl lepší výsledky pro identifikaci [35]. Zapojení dalších lidských vlastností je jedním ze způsobů posunutí úspěšnosti. Úspěšnost modelu rozpoznávání limitují data, která mají omezenou rozmanitost. Rozšíření trénovacích dat o data z odlišných podmínek by rozšířilo schopnosti systému.

Největší zpomalení systému vzniká během detekce osob. Nalezení lepšího způsobu detekce osob by zásadně zrychlilo běh systému. Možným způsobem by mohlo být například sloučení detekce a segmentace osob.

Vzhledem k zaměření systému na nenáročný provoz na širokém počtu zařízení je dalším relevantním cílem otestování a optimalizace na mobilních zařízeních.

Systém provádí pouze nejzákladnější rozpoznávání. Stále existuje mnoho situací, ve kterých nelze rozpoznání spolehlivě provést. Problém způsobuje překrytí rozpoznávaných osob. V případě pohybu dvou osob ve velké blízkosti, kdy vzájemně zasahují do svých bounding boxů, je další osoba zahrnutá v siluetě a narušuje proces rozpoznání. Problém odlišení jednotlivých instancí řeší instanční segmentace, která je ovšem složitější a pomalejší než použitá sémantická segmentace. Řešení problému vyžaduje nalezení způsobu, které bude splňovat požadavky systému. Oblečení a nesené břemeno ovlivňuje rozpoznávání. V některých případech může zamezit rozpoznávání. Pro systém je žádané odstranit nebo minimalizovat tyto vlivy. Řešením by mohlo být naučit neuronovou síť transformovat obrázky do formy nezatížené těmito vlivy.

Užitečným nástrojem, který by mohl rozšířit balíček se systémem, je skript pro vytváření binárních masek pomocí složitějšího a výkonnějšího modelu.

Po dokončení ostatních vylepšení může přijít na řadu potenciální rozšíření systému o ostatní úhly. Rozpoznávání z předního a zadního pohledu zůstává velkou výzvou. Tyto pohledy jsou výrazně odlišné od ostatních úhlů a poskytují o chůzi jiné informace. Rozšíření o přední a zadní pohled by vyžadovalo výrazné změny v jednotlivých částech systému.

Provedené testování se zabývalo pouze celkovými výsledky systému. Provedení testů zaměřených na specifické situace by odhalilo problematiku okolností a rozsah jejich vlivu.

Seznam literatury

[1] EVISON, Martin. Biometrics for Forensics. In: VAN TILBORG, Henk C. A. a Sushil JAJODIA, ed. Encyclopedia of Cryptography and Security [online]. Boston, MA: Springer US, 2011, 2011, s. 130-134 [cit. 2022-11-06]. ISBN 978-1-4419-5905-8. Dostupné z: doi:10.1007/978-1-4419-5906-5_731

[2] DANTCHEVA, Antitza, Carmelo VELARDO, Angela D'ANGELO a Jean-Luc DUGELAY. Bag of soft biometrics for person identification. *Multimedia Tools and Applications* [online]. 2011, **51**(2), 739-777 [cit. 2022-12-08]. ISSN 1380-7501. Dostupné z: doi:10.1007/s11042-010-0635-7

[3] JAIN, Anil K. a Arun ROSS. Introduction to Biometrics. In: JAIN, Anil K., Patrick FLYNN a Arun A. ROSS, ed. Handbook of Biometrics [online]. Boston, MA: Springer US, 2008, s. 1 -22 [cit. 2022-11-09]. ISBN 978-0-387-71040-2. Dostupné z: doi:10.1007/978-0-387-71041-9_1

[4] Zvláštní kategorie osobních údajů. Ministerstvo vnitra České republiky [online]. Praha: Ministerstvo vnitra České republiky, c2022 [cit. 2022-11-06]. Dostupné z: <https://www.mvcr.cz/gdpr/clanek/zvlastni-kategorie-osobnich-udaju.aspx>

[5] NEVRKLA, Luboš. Zpracování biometrických údajů zaměstnanců před a po GDPR. Epravo.cz [online]. 2020 [cit. 2022-11-07]. Dostupné z: <https://www.epravo.cz/top/clanky/zpracovani-biometrickych-udaju-zamestnancu-pred-a-po-gdpr-110639.html>

[6] Recital 18: Not Applicable to Personal or Household Activities. Intersoft Consulting: GDPR Info [online]. [cit. 2022-11-10]. Dostupné z: <https://gdpr-info.eu/recitals/no-18/>

[7] BOUCETTA, Rahma, Wiem ABDALLAH a Saloua BEL HADJ ALI. An Adaptive Control for the Path Tracking of an Active Leg Prosthesis. *International Journal of Robotics and Control Systems* [online]. 2021, 1 (2), 159-176 [cit. 2022-11-06]. ISSN 2775-2658. Dostupné z: doi:10.31763/ijrcs.v1i2.350

- [8] PARK, Gunwoo, Kyoung Min LEE a Seungbum KOO. Uniqueness of gait kinematics in a cohort study. *Scientific Reports* [online]. 2021, 11(1) [cit. 2022-11-07]. ISSN 2045-2322. Dostupné z: doi:10.1038/s41598-021-94815-z
- [9] YANG, Wencheng, Song WANG, Jiankun HU, Guanglou ZHENG a Craig VALLI. Security and Accuracy of Fingerprint-Based Biometrics: A Review. *Symmetry* [online]. 2019, 11(2) [cit. 2022-11-06]. ISSN 2073-8994. Dostupné z: doi:10.3390/sym11020141
- [10] PIRKER, Walter a Regina KATZENSCHLAGER. Gait disorders in adults and the elderly. *Wiener klinische Wochenschrift* [online]. 2017, **129**(3-4), 81-95 [cit. 2022-12-08]. ISSN 0043-5325. Dostupné z: doi:10.1007/s00508-016-1096-4
- [11] WAN, Changsheng, Li WANG a Vir V. PHOHA. A Survey on Gait Recognition. *ACM Computing Surveys* [online]. 2019, 51(5), 1 -35 [cit. 2022-11-06]. ISSN 0360-0300. Dostupné z: doi:10.1145/3230633
- [12] HADID, Abdenour, Mohammad GHAHRAMANI, Vili KELLOKUMPU, Matti PIETIKÄINEN, John BUSTARD a Mark NIXON. Can gait biometrics be Spoofed?. *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)* [online]. 2012, 21, 3280-3283 [cit. 2022-11-07]. Dostupné z: <https://projet.liris.cnrs.fr/imagine/pub/proceedings/ICPR-2012/media/files/1031.pdf>
- [13] BURT, Chris. Chinese police using Watrix gait biometrics as company plans major funding round. *Biometricupdate.com* [online]. 27 Feb 2019 [cit. 2022-11-07]. Dostupné z: <https://www.biometricupdate.com/201902/chinese-police-using-watrix-gait-biometrics-as-company-plans-major-funding-round>
- [14] SUN, Yiting. In China, a Store of the Future—No Checkout, No Staff. *MIT Technology Review* [online]. Cambridge (Massachusetts), 16 June 2017 [cit. 2022-11-07]. Dostupné z: <https://www.technologyreview.com/2017/06/16/151184/in-china-a-store-of-the-future-no-checkout-no-staff/>
- [15] TEEPE, Torben, Johannes GILG, Fabian HERZOG, Stefan HORMANN a Gerhard RIGOLL. Towards a Deeper Understanding of Skeleton-based Gait Recognition. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* [online]. IEEE, 2022, 2022, s. 1568-1576 [cit. 2022-11-06]. ISBN 978-1-6654-8739-9. Dostupné z: doi:10.1109/CVPRW56347.2022.00163

- [16] HIDALGO, Gines, Zhe CAO a subail. OpenPose. GitHub [online]. [cit. 2022-11-10]. Dostupné z: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>
- [17] FANG, Fred, jmydurant, Jeff LI a Lixin YANG. AlphaPose. GitHub [online]. [cit. 2022-11-10]. Dostupné z: <https://github.com/MVIG-SJTU/AlphaPose>
- [18] WANG, L., H. NING, T. TAN a W. HU. Fusion of Static and Dynamic Body Biometrics for Gait Recognition. IEEE Transactions on Circuits and Systems for Video Technology [online]. 2004, 14(2), 149-158 [cit. 2022-11-06]. ISSN 1051-8215. Dostupné z: doi:10.1109/TCSVT.2003.821972
- [19] HAN, J. a Bir BHANU. Individual recognition using gait energy image. IEEE Transactions on Pattern Analysis and Machine Intelligence [online]. 2006, 28(2), 316-322 [cit. 2021-04-19]. ISSN 0162-8828. Dostupné z: doi:10.1109/TPAMI.2006.38
- [20] ZHANG, Erhu, Yongwei ZHAO a Wei XIONG. Active energy image plus 2DLPP for gait recognition. Signal Processing [online]. 2010, 90(7), 2295-2302 [cit. 2022-11-06]. ISSN 01651684. Dostupné z: doi:10.1016/j.sigpro.2010.01.024
- [21] SAH, Shyamdeep a Sanjeeb PRASAD PANDAY. Model Based Gait Recognition Using Weighted KNN. Proceedings of IOE Graduate Conference (8th) [online]. 2020, 2020, 8, 8 [cit. 2022-11-06]. ISSN 2350-8914. Dostupné z: <http://conference.ioe.edu.np/ioegc8/papers/ioegc-8-131-80186.pdf>
- [22] KOVAČ, Jure, Vitomir ŠTRUC a Peter PEER. Frame-based classification for cross-speed gait recognition. Multimedia Tools and Applications [online]. 2019, 78(5), 5621-5643 [cit. 2022-11-06]. ISSN 1380-7501. Dostupné z: doi:10.1007/s11042-017-5469-0
- [23] KEHTARNAVAZ, Nasser, Matthias F. CARLSOHN, Mohammed AHMED, Naseer AL-JAWAD a Azhin SABIR. Gait recognition based on Kinect sensor [online]. In: 2014-5 - 15, 91390B- [cit. 2022-11-06]. Dostupné z: doi:10.1117/12.2052588
- [24] GHOSE, Abhishek. Support Vector Machine (SVM) Tutorial: Learning SVMs From Examples. KDnuggets [online]. [cit. 2022-12-07]. Dostupné z: <https://www.kdnuggets.com/2017/08/support-vector-machines-learning-svms-examples.html/3>

- [25] WANG, Xiuhui, Jun WANG a Ke YAN. Gait recognition based on Gabor wavelets and (2D)2PCA. *Multimedia Tools and Applications* [online]. 2018, 77(10), 12545-12561 [cit. 2022-11-06]. ISSN 1380-7501. Dostupné z: doi:10.1007/s11042-017-4903-7
- [26] MASOOD, Hajra, Humera FAROOQ a Muhammad Tariq MAHMOOD. An Appearance Invariant Gait Recognition Technique Using Dynamic Gait Features. *International Journal of Optics* [online]. 2021, 2021, 1 -15 [cit. 2022-11-06]. ISSN 1687-9392. Dostupné z: doi:10.1155/2021/5591728
- [27] DAS, Deepjoy a Sarat SAHARIA. Human Gait Analysis and Recognition Using Support Vector Machines. In: *Computer Science & Information Technology (CS & IT)* [online]. Academy & Industry Research Collaboration Center (AIRCC), 2014, 2014-7 -26, s. 187-195 [cit. 2022-11-06]. ISBN 9781921987083. Dostupné z: doi:10.5121/csit.2014.4725
- [28] CHOLLET, François. *Deep learning v jazyku Python: knihovny Keras, Tensorflow*. Přeložil Rudolf PECINOVSKÝ. Praha: Grada Publishing, 2019. Knihovna programátora (Grada). ISBN 978-80-247-3100-1.
- [29] SHIRAGA, Kohei, Yasushi MAKIHARA, Daigo MURAMATSU, Tomio ECHIGO a Yasushi YAGI. GEINet: View-invariant gait recognition using a convolutional neural network. In: *2016 International Conference on Biometrics (ICB)* [online]. IEEE, 2016, 2016, s. 1 -8 [cit. 2022-11-07]. ISBN 978-1 -5090-1869-7. Dostupné z: doi:10.1109/ICB.2016.7550060
- [30] LI, Chao, Xin MIN, Shouqian SUN, Wenqian LIN a Zhichuan TANG. DeepGait: A Learning Deep Convolutional Representation for View-Invariant Gait Recognition Using Joint Bayesian. *Applied Sciences* [online]. 2017, 7 (3) [cit. 2022-11-07]. ISSN 2076-3417. Dostupné z: doi:10.3390/app7030210
- [31] CHEN, Dong, Xudong CAO, Liwei WANG, Fang WEN a Jian SUN. Bayesian Face Revisited: A Joint Formulation. In: FITZGIBBON, Andrew, Svetlana LAZEBNIK, Pietro PERONA, Yoichi SATO a Cordelia SCHMID, ed. *Computer Vision – ECCV 2012* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, 2012, s. 566-579 [cit. 2022-11-07]. *Lecture Notes in Computer Science*. ISBN 978-3 -642-33711-6. Dostupné z: doi:10.1007/978-3 -642-33712-3 _41

- [32] SOKOLOVA, Anna a Anton KONUSHIN. Pose-based deep gait recognition. IET Biometrics [online]. 2019, 8 (2), 134-143 [cit. 2022-11-07]. ISSN 2047-4938. Dostupné z: doi:10.1049/iet-bmt.2018.5046
- [33] HADSELL, R., S. CHOPRA a Y. LECUN. Dimensionality Reduction by Learning an Invariant Mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06) [online]. IEEE, 2006, s. 1735-1742 [cit. 2022-11-10]. ISBN 0 -7695-2597-0. Dostupné z: doi:10.1109/CVPR.2006.100
- [34] SCHROFF, Florian, Dmitry KALENICHENKO a James PHILBIN. FaceNet: A unified embedding for face recognition and clustering. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [online]. IEEE, 2015, 2015, s. 815-823 [cit. 2022-11-10]. ISBN 978-1 -4673-6964-0. Dostupné z: doi:10.1109/CVPR.2015.7298682
- [35] TAKEMURA, Noriko, Yasushi MAKIHARA, Daigo MURAMATSU, Tomio ECHIGO a Yasushi YAGI. On Input/Output Architectures for Convolutional Neural Network-Based Cross-View Gait Recognition. IEEE Transactions on Circuits and Systems for Video Technology [online]. 2019, 29(9), 2708-2719 [cit. 2022-11-10]. ISSN 1051-8215. Dostupné z: doi:10.1109/TCSVT.2017.2760835
- [36] WANG, Chi-Feng. The Vanishing Gradient Problem. Towards Data Science [online]. 8 Jan 2019 [cit. 2022-12-07]. Dostupné z: <https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484>
- [37] WANG, Xiuhui a Wei Qi YAN. Human Gait Recognition Based on Frame-by-Frame Gait Energy Images and Convolutional Long Short-Term Memory. International Journal of Neural Systems [online]. 2020, 30(01) [cit. 2022-11-07]. ISSN 0129-0657. Dostupné z: doi:10.1142/S0129065719500278
- [38] KWABENA PATRICK, Mensah, Adebayo FELIX ADEKOYA, Ayidzoe ABRA MIGHTY a Baagyire Y. EDWARD. Capsule Networks – A survey. Journal of King Saud University - Computer and Information Sciences [online]. 2022, 34(1), 1295-1310 [cit. 2022-12-07]. ISSN 13191578. Dostupné z: doi:10.1016/j.jksuci.2019.09.014
- [39] MINAEE, Shervin, Yuri Y. BOYKOV, Fatih PORIKLI, Antonio J PLAZA, Nasser KEHTARNAVAZ a Demetri TERZOPOULOS. Image Segmentation Using Deep Learning:

A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence [online]. 1 -1 [cit. 2022-12-07]. ISSN 0162-8828. Dostupné z: doi:10.1109/TPAMI.2021.3059968

[40] YU, Shiqi, Haifeng CHEN, Qing WANG, Linlin SHEN a Yongzhen HUANG. Invariant feature extraction for gait recognition using only one uniform model. Neurocomputing [online]. 2017, 239, 81-93 [cit. 2022-11-07]. ISSN 09252312. Dostupné z: doi:10.1016/j.neucom.2017.02.006

[41] HU, Bingzhang, Yu GUAN, Yan GAO, Yang LONG, Nicholas LANE a Thomas PLOETZ. Robust Cross-view Gait Recognition with Evidence: A Discriminant Gait GAN (DiGGAN) Approach. Journal of the ACM [online]. 2018, August 2018, 37(4) [cit. 2022-11-07]. Dostupné z: doi:10.48550/arXiv.1811.10493

[42] XU, Zhaopeng, Wei LU, Qin ZHANG, Yuileong YEUNG a Xin CHEN. Gait recognition based on capsule network. Journal of Visual Communication and Image Representation [online]. 2019, 59, 159-167 [cit. 2022-11-07]. ISSN 10473203. Dostupné z: doi:10.1016/j.jvcir.2019.01.023

[43] PERSSON, Martin. Automatic gait recognition – using deep learning. Linköping, 2020. Diplomová práce. Linköpings universitet. Vedoucí práce Karl Holmquist.

[44] SAMMUT-BONNICI, Russell. Automated-Gait-Analysis. GitHub [online]. 2020 [cit. 2022-11-07]. Dostupné z: <https://github.com/RussellSB/automated-gait-analysis>

[45] MARGETA, Marian. DNN for gait recognition in TensorFlow. GitHub [online]. 2017 [cit. 2022-11-07]. Dostupné z: <https://github.com/marian-margeta/gait-recognition>

[46] LIANG, Kenneth. Human Identification Using Openpose. GitHub [online]. 2021 [cit. 2022-11-07]. Dostupné z: https://github.com/kenaimachine/HumanIdentification_Openpose

[47] CHOI, Seokeon. Skeleton-based Gait Recognition via Robust Frame-level Matching (RFM). GitHub [online]. 2019 [cit. 2022-11-07]. Dostupné z: <https://github.com/bismex/RFM>

[48] YU, Shiqi. OpenGait. GitHub [online]. [cit. 2022-11-10]. Dostupné z: <https://github.com/ShiqiYu/OpenGait>

[49] OpenCV Docs: Morphological Transformations [online]. [cit. 2022-11-29]. Dostupné z: https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html

[50] LIN, Tsung-Yi, Michael MAIRE, Serge BELONGIE, Lubomir BOURDEV, Ross GIRSHICK, James HAYS, Pietro PERONA, Deva RAMANAN, C. Lawrence ZITNICK a Piotr DOLLÁR. *Microsoft COCO: Common Objects in Context* [online]. 2014. Dostupné z: <http://arxiv.org/abs/1405.0312>. cite arxiv:1405.0312Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list

[51] KUCEV, Roman. Segmentation Full Body TikTok Dancing Dataset. Kaggle [online]. 2021 [cit. 2022-11-29]. Dostupné z: <https://www.kaggle.com/datasets/tapakah68/segmentation-full-body-tiktok-dancing-dataset>

[52] WANG, Liming, Jianbo SHI, Gang SONG a I-fan SHEN. Object Detection Combining Recognition and Segmentation. In: YAGI, Yasushi, Sing Bing KANG, In So KWEON a Hongbin ZHA, ed. *Computer Vision – ACCV 2007* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, s. 189-199 [cit. 2022-11-29]. Lecture Notes in Computer Science. ISBN 978-3-540-76385-7. Dostupné z: doi:10.1007/978-3-540-76386-4_17

[53] LIANG WANG, TIENIU TAN, HUAZHONG NING a WEIMING HU. Silhouette analysis-based gait recognition for human identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* [online]. 2003, 25(12), 1505-1518 [cit. 2022-11-29]. ISSN 0162-8828. Dostupné z: doi:10.1109/TPAMI.2003.1251144

[54] ZHENG, Shuai, Junge ZHANG, Kaiqi HUANG, Ran HE a Tieniu TAN. Robust view transformation model for gait recognition. In: 2011 18th IEEE International Conference on Image Processing [online]. IEEE, 2011, 2011, s. 2073-2076 [cit. 2022-11-29]. ISBN 978-1-4577-1303-3. Dostupné z: doi:10.1109/ICIP.2011.6115889

[55] SHIQI YU, DAOLIANG TAN a TIENIU TAN. A Framework for Evaluating the Effect of View Angle, Clothing and Carrying Condition on Gait Recognition. In: 18th International Conference on Pattern Recognition (ICPR'06) [online]. IEEE, 2006, 2006, s. 441-444 [cit. 2022-11-29]. ISBN 0-7695-2521-0. Dostupné z: doi:10.1109/ICPR.2006.67

[56] DAOLIANG TAN, KAIQI HUANG, SHIQI YU a TIENIU TAN. Efficient Night Gait Recognition Based on Template Matching. In: 18th International Conference on Pattern Recognition (ICPR'06) [online]. IEEE, 2006, 2006, s. 1000-1003 [cit. 2022-11-29]. ISBN 0-7695-2521-0. Dostupné z: doi:10.1109/ICPR.2006.478

- [57] TAKEMURA, Noriko, Yasushi MAKIHARA, Daigo MURAMATSU, Tomio ECHIGO a Yasushi YAGI. Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. *IP SJ Transactions on Computer Vision and Applications* [online]. 2018, 10(1) [cit. 2022-11-29]. ISSN 1882-6695. Dostupné z: doi:10.1186/s41074-018-0039-6
- [58] Persons Labeled [online]. 20 Nov 2020n. 1. [cit. 2022-11-29]. Dostupné z: <https://ecosystem.supervise.ly/projects/persons>
- [59] KUCEV, Roman. Supervise.ly Filtered Segmentation Person Dataset. Kaggle [online]. [cit. 2022-11-29]. Dostupné z: <https://www.kaggle.com/datasets/tapakah68/supervisely-filtered-segmentation-person-dataset>
- [60] RESHMA PRAKASH, Shet a Paras NATH SINGH. Object detection through region proposal based techniques. *Materials Today: Proceedings* [online]. 2021, 46, 3997-4002 [cit. 2022-12-07]. ISSN 22147853. Dostupné z: doi:10.1016/j.matpr.2021.02.533
- [61] HUI, Jonathan. Object detection: speed and accuracy comparison (Faster R -CNN, R -FCN, SSD, FPN, RetinaNet and YOLOv3). Medium [online]. 28 Mar 2018 [cit. 2022-11-18]. Dostupné z: <https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r -cnn-r -fcn-ssd-and-yolo-5425656ae359>
- [62] SUBRAMANYAM, Vineeth S. Non Max Suppression (NMS). Medium [online]. 20 Jan 2021 [cit. 2022-12-07]. Dostupné z: <https://medium.com/analytics-vidhya/non-max-suppression-nms-6623e6572536>
- [63] Object Detection on COCO test-dev [online]. [cit. 2022-11-18]. Dostupné z: <https://paperswithcode.com/sota/object-detection-on-coco>
- [64] MAGALHÃES, Sandro Augusto, Luís CASTRO, Germano MOREIRA, Filipe Neves DOS SANTOS, Mário CUNHA, Jorge DIAS a António Paulo MOREIRA. Evaluating the Single-Shot MultiBox Detector and YOLO Deep Learning Models for the Detection of Tomatoes in a Greenhouse. *Sensors* [online]. 2021, 21(10) [cit. 2022-11-18]. ISSN 1424-8220. Dostupné z: doi:10.3390/s21103569

- [65] REDMON, Joseph, Ali FARHADI, Santosh DIVVALA a Ross GIRSHICK. You Only Look Once: Unified, Real-Time Object Detection. Arxiv [online]. 2016 [cit. 2022-11-17]. Dostupné z: doi:arXiv.1506.02640
- [66] REDMON, Joseph a Ali FARHADI. YOLO9000: Better, Faster, Stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [online]. IEEE, 2017, 2017, s. 6517-6525 [cit. 2022-11-18]. ISBN 978-1-5386-0457-1. Dostupné z: doi:10.1109/CVPR.2017.690
- [67] Anchor Boxes for Object Detection. MathWorks [online]. [cit. 2022-11-18]. Dostupné z: <https://www.mathworks.com/help/vision/ug/anchor-boxes-for-object-detection.html>
- [68] REDMON, Joseph a Ali FARHADI. YOLOv3: An Incremental Improvement. Arxiv [online]. 8 Apr 2018 [cit. 2022-11-18]. Dostupné z: doi:<https://doi.org/10.48550/arXiv.1804.02767>
- [69] ST-CHARLES, Pierre-Luc, Guillaume-Alexandre BILODEAU a Robert BERGEVIN. SuBSENSE: A Universal Change Detection Method With Local Adaptive Sensitivity. IEEE Transactions on Image Processing [online]. 2015, 24(1), 359-373 [cit. 2022-12-07]. ISSN 1057-7149. Dostupné z: doi:10.1109/TIP.2014.2378053
- [70] NELSON, Joseph a Jacob SOLAWETZ. YOLOv4 vs YOLOv5. Roboflow: Blog [online]. Jun 12 2020 [cit. 2022-11-18]. Dostupné z: <https://blog.roboflow.com/yolov4-versus-yolov5/>
- [71] WANG, Chien-Yao, Alexey BOCHKOVSKIY a Hong-Yuan Mark LIAO. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. ArXiv [online]. 2022 [cit. 2022-11-18]. Dostupné z: doi:arXiv.2207.02696
- [72] BANERJEE, Amitabha. YOLOv5 vs YOLOv6 vs YOLOv7. Learn With a Robot [online]. [cit. 2022-11-18]. Dostupné z: <https://www.learnwitharobot.com/p/yolov5-vs-yolov6-vs-yolov7>
- [73] HungarianAlgorithm.com [online]. c2013–2022 [cit. 2022-12-07]. Dostupné z: <https://www.hungarianalgorithm.com/index.php>

- [74] BEWLEY, Alex, Zongyuan GE, Lionel OTT, Fabio RAMOS a Ben UPCROFT. Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP) [online]. IEEE, 2016, 2016, s. 3464-3468 [cit. 2022-11-19]. ISBN 978-1 -4673-9961-6. Dostupné z: doi:10.1109/ICIP.2016.7533003
- [75] WOJKE, Nicolai, Alex BEWLEY a Dietrich PAULUS. Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP) [online]. IEEE, 2017, 2017, s. 3645-3649 [cit. 2022-11-18]. ISBN 978-1 -5090-2175-8. Dostupné z: doi:10.1109/ICIP.2017.8296962
- [76] FRANKLIN, William. Kalman Filter Explained Simply. The Kalman Filter [online]. 31 Dec 2020 [cit. 2022-11-19]. Dostupné z: <https://thekalmanfilter.com/kalman-filter-explained-simply>
- [77] SANDLER, Mark, Andrew HOWARD, Menglong ZHU, Andrey ZHMOGINOV a Liang-Chieh CHEN. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition [online]. IEEE, 2018, 2018, s. 4510-4520 [cit. 2022-12-07]. ISBN 978-1 -5386-6420-9. Dostupné z: doi:10.1109/CVPR.2018.00474
- [78] POUDEL, Rudra P K, Stephan LIWICKI a Roberto CIPOLLA. Fast-SCNN: Fast Semantic Segmentation Network [online]. arXiv, 2019 [cit. 2022-11-21]. Dostupné z: doi:10.48550/ARXIV.1902.04502
- [79] ADALOGLOU, Nikolas. Intuitive Explanation of Skip Connections in Deep Learning. AI Summer [online]. 23 Mar 2020 [cit. 2022-11-22]. Dostupné z: <https://theaisummer.com/skip-connections>
- [80] ZHAO, Hengshuang, Jianping SHI, Xiaojuan QI, Xiaogang WANG a Jiaya JIA. Pyramid Scene Parsing Network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [online]. IEEE, 2017, 2017, s. 6230-6239 [cit. 2022-11-22]. ISBN 978-1 -5386-0457-1. Dostupné z: doi:10.1109/CVPR.2017.660
- [81] REINHARD, E., M. ADHIKHMEN, B. GOOCH a P. SHIRLEY. Color transfer between images. IEEE Computer Graphics and Applications [online]. 21(4), 34-41 [cit. 2022-11-24]. ISSN 02721716. Dostupné z: doi:10.1109/38.946629

[82] PolynomialDecay. Keras [online]. [cit. 2022-11-25]. Dostupné z: https://keras.io/api/optimizers/learning_rate_schedules/polynomial_decay/

[83] XU, Chi, Yasushi MAKIHARA, Xiang LI, Yasushi YAGI a Jianfeng LU. Cross-View Gait Recognition Using Pairwise Spatial Transformer Networks. IEEE Transactions on Circuits and Systems for Video Technology [online]. 2021, 31(1), 260-274 [cit. 2022-11-26]. ISSN 1051-8215. Dostupné z: doi:10.1109/TCSVT.2020.2975671

Seznam obrázků

Obrázek 1: Cyklus chůze a jeho fáze [10]	8
Obrázek 2: Ukázka reprezentace pomocí Sketetonu [15].....	12
Obrázek 3: Ukázka GEI [19]	13
Obrázek 4: Ukázka AEI [20]	13
Obrázek 5: Lineárně oddělitelné třídy, uprostřed optimální nadrovina, margin v modrém a vyznačené pomocné vektory [24]	15
Obrázek 6: Průběh činnosti konvoluční vrstvy [28].....	17
Obrázek 7: Proces identifikace	25
Obrázek 8: Zjednodušený UML diagram systému.....	27
Obrázek 9: Architektura Fast-SCNN sítě [78]	38
Obrázek 10: Vlevo původní obrázek [52], vpravo obrázek po změně pozadí.....	39
Obrázek 11: Seznam augmentací, parametr p označuje pravděpodobnost výskytu.....	40
Obrázek 12: Ukázka výstupy Fast-SCNN.....	41
Obrázek 13: Vlevo fáze Mid, vpravo fáze Heel strike.....	41
Obrázek 14: Síť pro klasifikaci fáze kroku.....	42
Obrázek 15: Síť pro klasifikaci směru	44
Obrázek 16: a) 2in b) diff [35]	46

Seznam tabulek

Tabulka 1: Matice záměn klasifikace fáze kroku	43
Tabulka 2: Matice záměn klasifikace směru chůze na validačním datasetu	45
Tabulka 3: Matice záměn klasifikace směru chůze na testovacím datasetu.....	45
Tabulka 4: Výsledky vlastní implementace diff2in na validačních a testovacích datasetech s prahem, který dosahuje EER.....	48
Tabulka 5: Výsledky vlastní implementace diff2in na validačních datasetech s prahem na úrovni, kde poměr FN a FP je 6 : 1	49
Tabulka 6: Výsledky diff2in s prahem 1,6776 na testovacích datasetech	49
Tabulka 7: Matice záměn verifikace osob systémem na testovacím zařízení.....	50
Tabulka 8: Matice záměn verifikace osob systémem ve 30 FPS	50
Tabulka 9: Výsledky identifikace na testovacím zařízení. Neidentifikováno znamená, že byla osoba určena jako neznámá.	50
Tabulka 10: Výsledky identifikace ve 30 FPS. Neidentifikováno znamená, že byla osoba určena jako neznámá	50

Přílohy

V souboru s přílohami se nachází:

- Kompletní implementace systému
- README soubor, který obsahuje návod na zprovoznění, popis struktury balíčku a stručný popis používání
- UML class diagram systému
- Schéma sítě pro určení fáze směru
- Schéma sítě pro určení směru