



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**

DEPARTMENT OF INTELLIGENT SYSTEMS

**SLEDOVÁNÍ POHYBU NÁVŠTĚVNÍKŮ MUZEJNÍCH  
EXPOZIC**

MONITORING THE MOVEMENT OF VISITORS IN MUSEUM EXHIBITIONS

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**MATEJ VISKUPIČ**

**VEDOUcí PRÁCE**

SUPERVISOR

Prof. Ing., Dipl.-Ing. MARTIN DRAHANSKÝ, Ph.D.

BRNO 2021

## Zadání bakalářské práce



Student: **Viskupič Matej**  
Program: Informační technologie  
Název: **Sledování pohybu návštěvníků muzejních expozic**  
**Monitoring the Movement of Visitors in Museum Exhibitions**  
Kategorie: Zpracování obrazu

### Zadání:

1. Prostudujte literaturu týkající se detekce a sledování pohybujících se objektů zájmu ve videosekvenčních souborech.
2. Navrhněte algoritmus pro sledování pohybu návštěvníků v muzejních expozicích. Navrhněte rovněž vizualizaci jejich pohybu v půdorysné projekci.
3. Vámi navržené řešení z bodu 2 prakticky implementujte.
4. Proveďte experimenty na reálných datech a dosažené výsledky zhodnoťte.

### Literatura:

- DRAHANSKÝ, Martin; MACEK, Ivo; GOLDMANN, Tomáš. Monitoring of visitors in museum exhibitions. *Journal of the National Museum (Prague), Natural History Series*, 2020, 189.
- LANIR, Joel, et al. Visualizing museum visitors' behavior: Where do they go and what do they do there?. *Personal and Ubiquitous Computing*, 2017, 21.2: 313-326.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

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

Vedoucí práce: **Drahanský Martin, prof. Ing., Dipl.-Ing., Ph.D.**

Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.

Datum zadání: 1. listopadu 2021

Datum odevzdání: 11. května 2022

Datum schválení: 4. listopadu 2021

## Abstrakt

Cielom tejto práce je navrhnúť riešenie problému monitorovania návštevníkov múzejných expozícií. Na rozdiel od už existujúcich metód je tento problém riešený použitím kamerovej technológie. Tento prístup si vyžiadal vyriešenie troch podproblémov: (1.) detekciu ľudí v kamerovom zázname za využitia konvolučnej neurónovej siete, (2.) presné určenie polohy zaznamenaných osôb v rámci monitorovaného priestoru za pomoci konfigurácie pozície kamier, (3.) identifikáciu a sledovanie zaznamenaných osôb. Výstupom navrhovaného riešenia je tepelná mapa, vykresľujúca najnavštevovanejšie miesta, mapa trajektórií návštevníkov znázorňujúca ich pohyb a štatistika návštevnosti jednotlivých expozícií. Tieto výstupy môžu prispieť k lepšiemu vyhodnocovaniu potrieb návštevníkov a k efektívnejšiemu výberu alebo rozmiestneniu exponátov.

## Abstract

The aim of this work is to propose a new system of monitoring visitors in museums. In contrast to existing methods, the problem is solved here only using camera technology. This requires addressing three sub-problems: (1.) detection of visitors in camera streams using a convolutional neural network; (2.) camera configuration to exactly determine the position of the detected persons within the monitoring area; and (3.) identification and tracking the detected persons. The outcome of the proposed solution is the heatmap of most visited places, the map of visitor trajectories and the statistic of visits for individual exhibits. This monitoring method can contribute towards improved evaluation of visitor experience and more effective selection and positioning of the exhibits.

## Kľúčové slová

identifikácia osôb, sledovanie osôb, tepelná mapa, monitorovanie múzea

## Keywords

person identification, person tracking, heatmap, museum monitoring

## Citácia

VISKUPIČ, Matej. *Sledování pohybu návštěvníků muzejních expozic*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Prof. Ing., Dipl.-Ing. Martin Drahanský, Ph.D.

# Sledování pohybu návštěvníků muzejních expozic

## Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Dra-hanského. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....

Matej Viskupič

9. mája 2022

## Podakovanie

Chcel by som poďakovať môjmu školiteľovi prof. Ing. Martinovi Dra-hanskému Ph.D. za jeho pomoc pri písaní práce a jeho odborné rady pri implementovaní riešenia.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Zhrnutie existujúcich technológií</b>	<b>3</b>
2.1	Sledovanie návštevníkov bez využitia kamerovej technológie . . . . .	3
2.2	Identifikácia ľudí v kamerovom signále . . . . .	5
2.3	Translácia medzi súradnicovým systémom kamery a reálneho sveta . . . . .	10
2.4	Sledovanie ľudí naprieč snímkami a kamerami . . . . .	12
2.5	Priradenie detekcií existujúcim osobám . . . . .	13
<b>3</b>	<b>Návrh</b>	<b>19</b>
3.1	Kalibrácia kamier - inicializácia . . . . .	19
3.2	Detekcia a re-identifikácia osôb . . . . .	20
<b>4</b>	<b>Implementácia</b>	<b>22</b>
4.1	Implementácia kalibrácie kamier . . . . .	22
4.2	Implementácia riadiaceho systému . . . . .	24
4.3	Identifikácia a sledovanie osôb . . . . .	27
4.4	Generovanie výstupu . . . . .	29
4.5	Užívateľské rozhranie . . . . .	32
<b>5</b>	<b>Testovanie</b>	<b>33</b>
5.1	Využitie datasety . . . . .	33
5.2	Výsledky testovania na datasete WILDTRACK . . . . .	34
5.3	Výsledky testovania na vlastnom datasete . . . . .	37
<b>6</b>	<b>Záver</b>	<b>38</b>
	<b>Literatúra</b>	<b>39</b>

# Kapitola 1

## Úvod

Pre odvetvia, ktorých hlavným prvkom je interakcia s ľuďmi, je kľúčovým nástrojom zisťovanie spätnej väzby od svojich užívateľov. Na vytvorenie tejto spätnej väzby je potrebné získať údaje od užívateľov. Tieto údaje je možné získavať rôznymi spôsobmi ale pre najefektívnejšie vyhodnocovanie a najrýchlejšie zdokonaľovanie poskytovaných služieb je ich potrebné získavať rýchlo, presne a lacno. Múzeá či galérie patria medzi tieto odvetvia, pre ktoré je spätná väzba od svojich užívateľov dôležitá. Rovnako ako sú pre sociálne siete veľmi dôležité informácie o tom, ktoré príspevky používateľov zaujali, tak aj pre múzeá sú veľmi hodnotné informácie o stave ich expozícií. Odpovede na otázky ako sú: „Ktoré expozície zaujali najviac ľudí?“ alebo „Kadiaľ sa ľudia pohybovali?“

Využívanie ľudí na zber týchto dát môže byť nákladné a zároveň predstavuje mrhanie ľudskými zdrojmi. Predovšetkým je využívanie ľudí na túto prácu nevýhodné ak, je potrebné sledovať väčšie alebo viaceré objekty zároveň. Niektoré z existujúcich riešení na túto úlohu využívajú špeciálne zariadenia. Dokážu s vysokou presnosťou určiť polohu návštevníka avšak prichádzajú s dvomi značnými nevýhodami. Prvou z nich je nutnosť nosenia tohto špecializovaného zariadenia návštevníkom. Týmto zťažujú návštevníkov a ovplyvňujú ich celkový zážitok z múzea. Druhou nevýhodou predstavuje požiadavka na vytvorenie veľkej siete detekčných zariadení naprieč sledovaným objektom, kvôli nepostačujúcemu dosahu aktuálne existujúcich zariadení.

Táto práca navrhuje riešenie problému získavania spätnej väzby od návštevníkov za pomoci kamerovej technológie. Navrhuje využitie popredných metód, ktoré vo svojich oblastiach dosahujú najlepšie výsledky, na sledovanie správania návštevníkov a na jeho základe generovanie spätnej väzby. Naše riešenie na rozdiel od už existujúcich riešení v žiadnej miere nezťažuje návštevníka nosením špecializovaných zariadení. Kamery oproti zariadeniam použitých v existujúcich riešeniach majú podstatne väčší dosah detekcie a identifikácie človeka. Vďaka tomu navrhované riešenie vyžaduje menej rušivých elementov v múzeu a jednoduchšiu celkovú infraštruktúru senzorov.

Problém detekcie a sledovania návštevníkov je bližšie priblížený v nasledujúcich kapitolách, ktoré sú členené do dvoch celkov. Prvým z nich je teoretická časť pozostávajúca z kapitoly 2. Je v nej zhrnutý doterajší výskum a aktuálny postup technológie v oblastiach významných pre túto prácu. Druhý z celkov predstavuje praktickú časť, v ktorej je opísaný samotný systém. V kapitole 3 je predstavený návrh systému, ktorý stavia na doterajších výskumoch. Popis podrobností implementácie systému je bližšie popísaný v kapitole 4. Táto implementácia je následne otestovaná a výsledky testov zhodnotené v kapitole 5. Z týchto výsledkov a doposiaľ zistených informácií je v kapitole 6 vyvodený záver.

## Kapitola 2

# Zhrnutie existujúcich technológií

V nasledujúcich sekciách tejto kapitoly je priblížený stav doterajšieho výskumu v oblastiach, ktoré sú významné pre túto prácu. Nejedná sa o encyklopedický popis daných disciplín ale o zhrnutie ich najmodernejších a najpodstatnejších častí.

### 2.1 Sledovanie návštevníkov bez využitia kamerovej technológie

Pre každé z odvetví, ktorého hlavným aspektom je interakcia s ľuďmi, sú kľúčové informácie o preferenciách svojich zákazníkov a ich správani. Medzi tieto odvetvia patria aj múzeá a galérie. Rovnako ako web-stránky zbierajú údaje o pohybe kurzora myši či štatistiky o najviac kliknutých elementoch na stránke, tak aj múzeá sledujú svojich návštevníkov. Z údajov z tohto sledovania sa snažia identifikovať najpríťažlivejšie exponáty či najvhodnejšie miesta pre rozmiestnenie exponátov a jednotlivých informačných, bezpečnostných či sociálnych prvkov.

Pri sledovaní návštevníkov múzea štúdia [4] popisuje nasledovné metriky pre hodnotenie jednotlivých exponátov v expozícii:

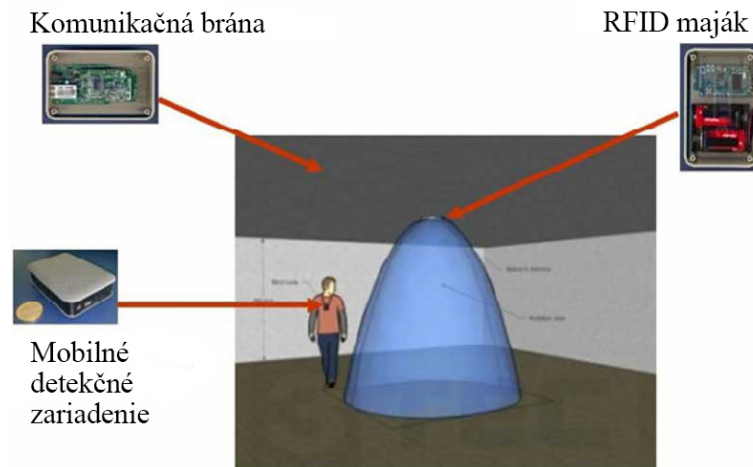
- **Príťahujúca sila:** vyjadruje relatívny počet ľudí, ktorý sa zastavili pred objektom/exponátom počas ich návštevy. Vypočíta sa ako podiel ľudí, ktorý sa pristavili pri objekte, celkovým počtom ľudí, ktorý návštívili múzeum.

$$\text{Príťahujúca sila} = \frac{\text{Počet ľudí, ktorý sa pristavili}}{\text{Celkový počet návštevníkov}}$$

- **Udržujúca sila:** meria priemerný čas strávený pred informačným/komunikačným prvkom (napríklad panel, video, popisok ...). Počíta sa ako podiel priemerného času pristavenia sa návštevníkov, „potrebným“ časom na prečítanie si prvku. „Potrebný“ čas je určený odborníkmi a predstavuje čas potrebný na vstrebanie informácií. Čím bližšie je táto hodnota k 1, tým väčšmi si element vie udržať pozornosť návštevníkov.

$$\text{Udržujúca sila} = \frac{\text{Priemerný čas pristavenia}}{\text{Potrebný čas na pristavenie}}$$

Väčšina z výskumov, zaoberajúcich sa touto problematikou, bolo vykonávaných takzvanou metódou „ceruzky a papiera“, kde sa výskumníci alebo personál múzea fyzicky nachádzal v monitorovaných priestoroch a ručne si zapisoval správanie návštevníkov. Ďalšou



Obr. 2.1: Vizualizácia prístupu štúdie [20] k sledovaniu návštevníkov za pomoci technológie RFID. Na obrázku je znázornený dosah RFID majáku formou modrého kužeľu<sup>1</sup>.

metódou je takzvané „front-end vyhodnotenie“, kde sú návštevníkom položené otázky pred vstupom do monitorovanej oblasti a následne po jej opustení [8]. Tento prístup zberu dát vyžaduje od návštevníka aktívne zapojenie sa a tým priamo zasahuje do jeho zážitku z daného múzea. Rovnako sa tento prístup náročne škáluje na väčšie monitorované oblasti, keďže na zber dát vyžaduje ľudskú pracovnú silu.

Boli skúmané aj iné prístupy k sledovaniu návštevníkov a to za využitia externého zariadenia, taktiež nazývaného lokátor, ktoré má návštevník stále so sebou počas jeho návštevy monitorovanej oblasti. Týmto zariadením môže byť napríklad jeho chytrý mobilný telefón s technológiou Wi-Fi [31]. Naprieč sledovanou oblasťou je rozmiestnená sieť niekoľkých prístupových bodov. Návštevníkov chytrý telefón za pomoci špeciálnej aplikácie sleduje silu prijatého signálu (RSS Received Signal Strength) z jednotlivých prístupových bodov a tým určuje jeho približnú polohu. Túto technológiu slúžiacu na identifikovanie populárnych expozícií už využívajú niektoré z väčších múzeí, menovite Národná Galéria a Národné Historické múzeum v Londýne [8].

Lokátor nemusí byť len návštevníkov chytrý telefón. Pre rôzne účely môže byť vhodné použiť rôzne technológie, napríklad RFID (Radio Frequency Identification), UWB (Ultra-Wideband) či GPS (Global Positioning System). Každá z týchto technológií ponúka svoje vlastné výhody a nevýhody. GPS je vhodné použiť na rozsiahlych vonkajších miestach, kde nepotrebujeme zisťovať polohu s vysokou presnosťou ako sú napríklad zoologické záhrady. Naopak štúdia [20] sa zaoberá technológiou RFID, ktorá vie za pomoci veľkého počtu rádiových majákov presne určiť návštevníkovu polohu. Po tom čo sa lokátor dostane do dosahu signálu majáka navzájom začnú komunikovať, čím sa presne určí poloha lokátora a tým aj návštevníka. Zóna dosahu a jednotlivé prvky systému sú znázornené na obrázku 2.1. Toto využitie RFID lokátorov je však obmedzené silou ich signálu a teda maximálnou vzdialenosťou v akej dokážu medzi sebou komunikovať a teda určovať polohu. Rovnako je návštevník zaťažovaný nosením lokátora počas celej jeho návštevy monitorovanej oblasti.

<sup>1</sup>Obrázok bol preložený do slovenčiny a pochádza zo štúdie [20]



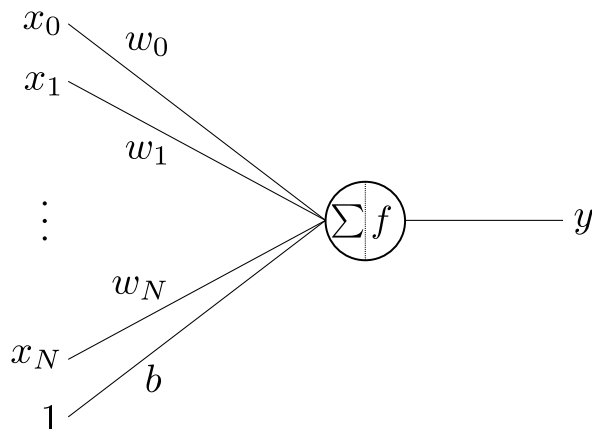
## 2.2 Identifikácia ľudí v kamerovom signále

Je prirodzenou ľudskou schopnosťou pozrieť sa na obrázok a okamžite v ňom identifikovať známe objekty, určiť ich umiestnenie, tvar a farbu. Túto schopnosť detekcie objektov sa snažíme naučiť stroje. Dnes sa táto detekcia objektov v obrazovom zázname v prevažnej miere vykonáva za pomoci neurónových sietí. Kľúčovými výzvami stále zostávajú presnosť, rýchlosť, cena a zložitosť daného riešenia [9].

### 2.2.1 Princíp fungovania neurónových sietí

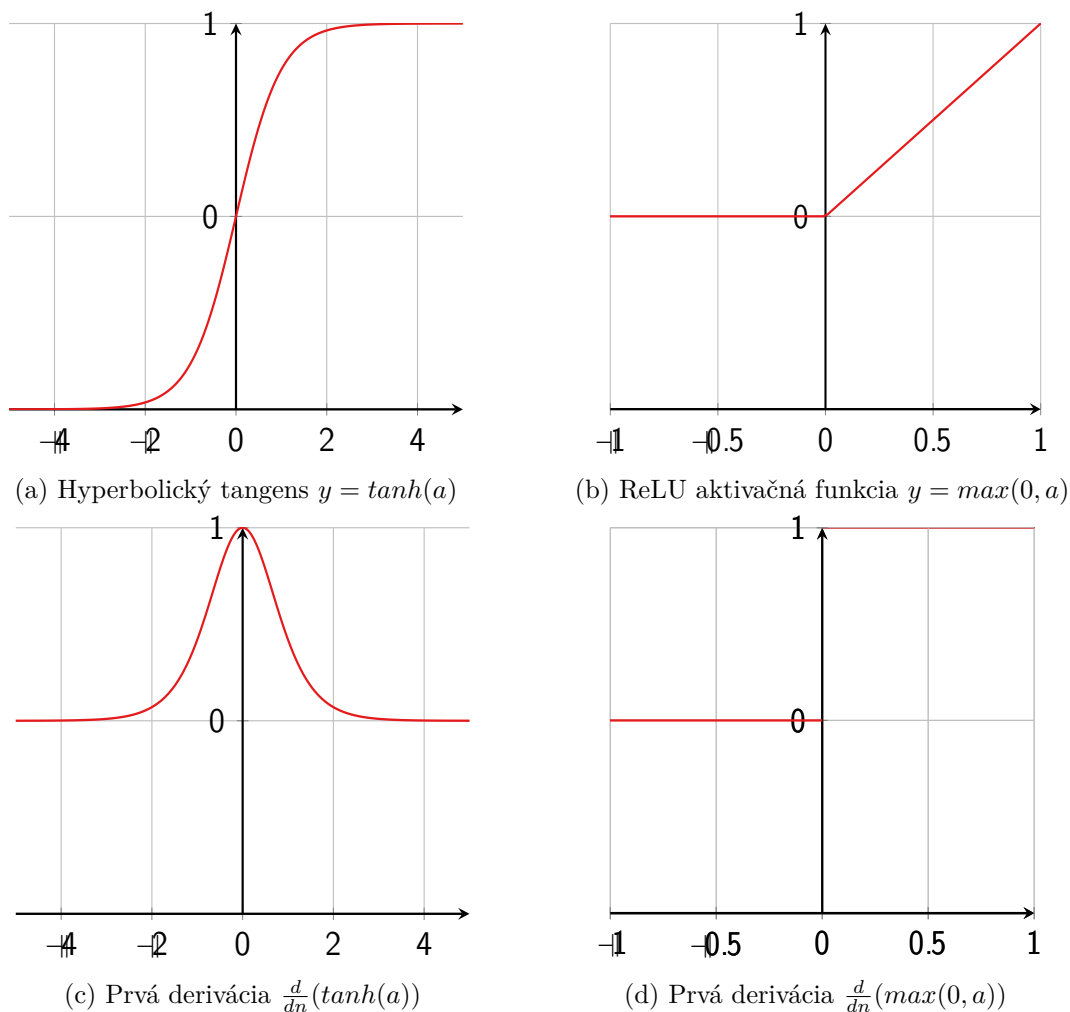
Neurónové siete sú softvérovou implementáciou našich mozgov. Snažia sa imitovať a dôkladne napodobňovať funkciu biologických neurónových sietí, z ktorých sa skladajú naše mozgy. Bez zachádzania do biologických podrobností môžeme povedať, že naše mozgy pozostávajú z veľkého počtu navzájom prepojených neurónov, ktoré pripomínajú spínače. Svoj stav zapnutia menia na základe sily svojich chemických či elektrických vstupov. Pričom stav jedného neuróna slúži ako vstup pre ďalšie neuróny [16].

Neuróny v umelých neurónových sieťach sú veľmi podobné svojim biologickým náprotivníkom. Pozostávajú zo vstupov, váh a bias. Schéma jednoduchého neurónu sa nachádza na obrázku 2.2. Každý zo vstupov neurónu je vynásobený svojou váhou, následne sú tieto hodnoty sčítané, je k nim pripočítaný bias a táto výsledná hodnota slúži ako vstup do aktivačnej funkcie. Výstupom neurónu je výstup aktivačnej funkcie. Táto funkcia vyjadruje to, či a ako veľmi je neurón aktivovaný. Väčšinou sa používa funkcia sigmoid, ktorej výhoda je plynulý prechod medzi okrajovými hodnotami a jej ohraňovanie výstupu na intervale  $\langle 0,1 \rangle$ . Avšak používajú sa aj iné aktivačné funkcie, tie najčastejšie používané možno vidieť na obrázku 2.3.



Obr. 2.2: Ilustrácia neurónu.  $x_0$  až  $x_N$  predstavujú vstupy,  $w_0$  až  $w_N$  váhy jednotlivých vstupov,  $b$  bias neurónu a  $f$  aktivačná funkcia. Výstup neurónu predstavuje  $y^2$ .

<sup>2</sup>Obrázok pochádza zo štúdie [16]



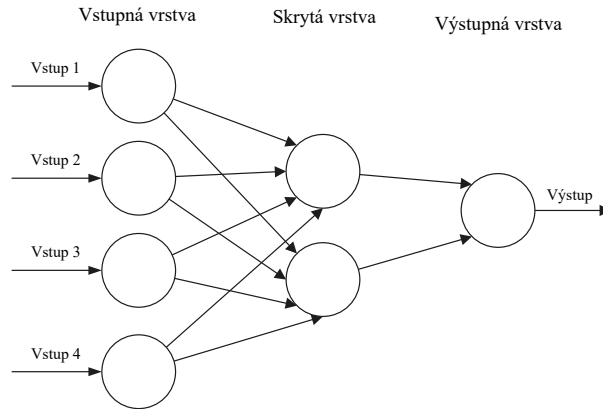
Obr. 2.3: Často používané aktivačné funkcie<sup>3</sup>.

Neurónová sieť pozostáva z mnohých neurónov usporiadaných do vrstiev. Každý neurón patrí iba do jednej vrstvy a je napojený na neuróny predchádzajúcej a nasledujúcej vrstvy. Táto štruktúra môže nadobúdať rôzne tvary, teda líšiť sa počtom vrstiev a počtom neurónov v týchto vrstvách. Jednoduchú štruktúru možno vidieť na obrázku 2.4. V klasickej umelej neurónovej sieti (ANN) rozlišujeme 3 rôzne vrstvy:

- *Vstupná vrstva* (Input layer), cez ktorú dáta vstupujú do siete.
- *Skryté vrstvy* (Hidden layer), ktoré nie sú ani vstupné ani výstupné.
- *Výstupná vrstva* (Output layer), z ktorej získavame pravdepodobnosti jednotlivých výstupov.

V našich mozgoch sa opakovaným vykonávaním určitej činnosti alebo opakovaním určitých znalostí zosilňujú prepojenia medzi jednotlivými neurónmi. Týmto procesom sa zvyšuje šanca na aktiváciu tých istých neurónov pri predložení rovnakých alebo podobných vstupov

<sup>3</sup>Obrázok pochádza zo štúdie [16].



Obr. 2.4: Štruktúra jednoduchej umelej neurónovej siete. Obsahuje jednu vstupnú vrstvu, jednu skrytú vrstvu a jednu výstupnú vrstvu<sup>4</sup>.

v budúcnosti. Tento proces opakovania činnosti sa nazýva učenie a tvorí rovnako dôležitú časť aj pri umelých neurónových sieťach. Tu ho rozdeľujeme na dve kategórie:

- **Učenie s dozorom** Pri tomto spôsobe máme k dispozícii vstupné dáta spolu so správnymi výstupmi. Umelej neurónovej sieti sú predložené vstupné dáta a na základe porovnávania očakávaných výstupných dát s výstupnými dátami neurónovej siete zasahujeme do jej fungovania a meníme ho.
- **Učenie bez dozoru** Tento spôsob sa líši tým, že nemáme k dispozícii správne výstupy. Nechávame neurónovú sieť pracovať samu na rozoznávaní nových vzorov a informácií, ktoré boli predtým nedetegované. Úspech neurónovej siete je väčšinou určený tým, či sa podarilo znížiť alebo zvýšiť pridruženú cenovú funkciu (associated cost function).

### 2.2.2 Princíp fungovania CNN

Konvolučné neurónové siete (Convolutional Neural Network - CNN) sú rovnako ako umelé neurónové siete zložené z neurónov uložených do vrstiev. Avšak tu ich podobnosť končí. CNN sú vhodné na prácu s obrazovými dátami preto aj ich architektúra bola optimalizovaná na túto úlohu. Narozdiel od klasických neurónových sietí, neuróny v CNN sú prepojené len na časť neurónov v predchádzajúcej vrstve. Toto má za dôsledok zredukovanie parametrov v neurónovej sieti [1]. Princíp zoskupenia neurónov do viacerých vrstiev zostal oproti klasickým neurónovým sieťam nezmenený avšak mení sa funkcia jednotlivých vrstiev. Konkrétne sa CNN skladá z nasledujúcich vrstiev [9]:

*Vstupná vrstva* plní úlohu inicializácie vstupných obrazových dát a normalizácie ich škály do hodnôt  $[0,1]$ . Využíva Principal Components Analysis (PCA) na degradáciu a dekorreláciu dát pre lepšie zameranie sa na niekoľko kľúčových faktorov.

*Konvolučná vrstva* je jadrom celej CNN. Používa konvolučné jadro ako filter, ktorý sa postupne aplikuje na pôvodné obrazové dáta. Násobením dát z predchádzajúcej vrstvy filtrom a ich následným sčítaním dostávame konvolučný výsledok.

*Pooling vrstva* je použitá na zníženie dimenzií výsledkov predchádzajúcich vrstiev. Existujú tri typy pooling: všeobecný pooling, prekrývajúci sa pooling a Spatial Pyramid

<sup>4</sup>Obrázok bol preložený do slovenčiny a pochádza zo štúdie [27]

Pooling (SPP). Veľmi často používané metódy všeobecného pooling-u sú extrahovanie maximálnej hodnoty z oblasti a počítanie priemernej hodnoty v danej oblasti.

*Plne prepojená vrstva* často predstavuje poslednú vrstvu. Pracuje na rovnakom princípe ako vrstvy v klasickej umelej neurónovej sieti. Na jej výstupe sa nachádza skóre pre jednotlivé klasifikácie.

### 2.2.3 Klasifikácia CNN

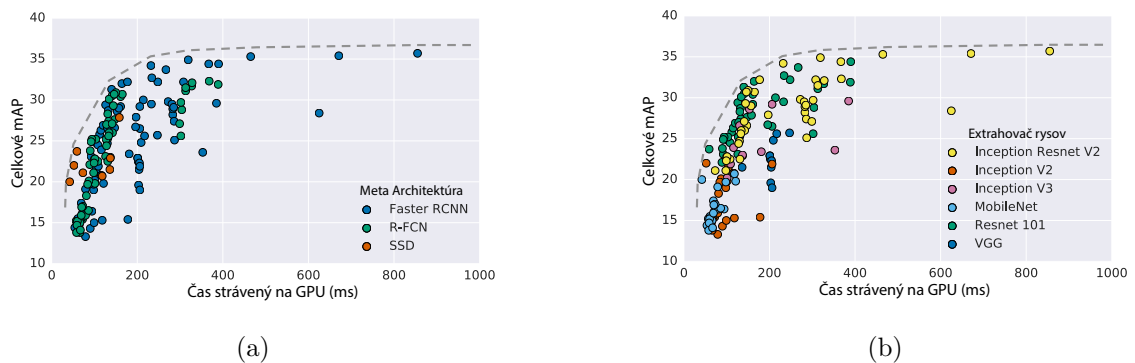
Medzi najmodernejšie detektory objektov založené na technológii CNN patria napríklad Faster R-CNN, R-FCN, SSD, Multibox a YOLO. Tieto detektory by sme mohli rozdeliť podľa štúdie [15] na niekoľko hlavných, najnovších architektur. Menovite hovorí o architektúrach SSD (Single Shot Multibox Detector), Faster R-CNN a R-FCN (Region-based Fully Convolutional Networks).

**Single Shot Detector (SSD)** Tento termín sa vo všeobecnosti používa na označovanie architektúr, ktoré používajú jednu doprednú (feed-forward) konvolučnú neurónovú sieť na priame zisťovanie tried a ofsetov bez nutnosti využiť druhú úroveň klasifikácie pre jednotlivé návrhy.

**Faster R-CNN** Pri tejto architektúre detekcia prebieha v dvoch krokoch. Prvý krok predstavuje aplikovanie konvolučnej neurónovej siete nazývanej Region Proposal Network (RPN), ktorá spracuje vstupný obraz a deteguje regióny záujmu, ktoré sú následne postúpené druhej časti. V druhej fáze je využitý detektor Fast R-CNN, ktorý z prevzatých regiónov určuje triedu a spresňuje ohraničujúce boxy špecifické pre danú triedu. Tento dvojkrokový prístup si nevyžaduje opakované spúšťanie extrahovača rysov pre oba kroky, čím sa vyhýba zbytočnému dvojitému výpočtu. Avšak je potrebné vykonať určitú časť výpočtov pre každý región. Práve z tohto dôvodu čas behu závisí od množstva regiónov navrhnutých prvým krokom (RPN). Vďaka zdokonaleniu prvého kroku je Faster R-CNN niekoľkonásobne rýchlejšia ako Fast R-CNN [15].

**Region-based Fully Convolutional Networks (R-FCN)** Táto metóda sa veľmi podobá predošlej metóde, Faster R-CNN. Hlavným rozdielom je miesto, kde sa vykonáva vystrihovanie rysov. Oproti Faster R-CNN, kde sa vystrihovanie dialo na tej istej vrstve z akej boli navrhnuté regióny, v tejto metóde sa vystrihovanie vykonáva až na poslednej vrstve pred samotným určovaním tried. Tento prístup oddalovania vystrihovania až na poslednú vrstvu minimalizuje množstvo výpočtov, ktoré musia byť vykonané na každom regione. Tým pádom dosahuje ešte rýchlejšie časy výpočtov pri zachovaní podobnej presnosti ako Faster R-CNN [15].

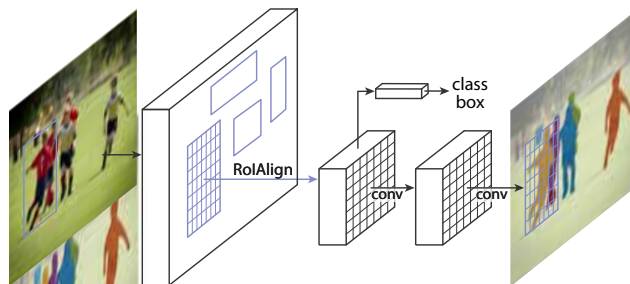
Na obrázku 2.5 je vidieť porovnanie rýchlosti týchto jednotlivých architektur spolu s extrahovačmi rysov.



Obr. 2.5: Porovnanie rýchlosti jednotlivých architektúr 2.5a a extrahovačov rysov 2.5b. V grafe je znázornená stredná hodnota priemernej presnosti (mAP) voči reálnemu času strávenému na GPU<sup>5</sup>.

## 2.2.4 Mask R-CNN

Konvolučná sieť Mask R-CNN [11] stavia na základoch metódy Faster R-CNN, ktorú rozširuje o vetvu určujúcu segmentačné masky na každom regióne záujmu (Region of Interest - RoI). Táto vetva sa vykonáva paralelne s existujúcou vetvou pre klasifikáciu a zisťovanie ohraničujúcich boxov (bounding-box). Tento proces je vyobrazený na obrázku 2.6. V princípe sa jedná o intuitívne rozšírenie neurónovej siete Faster R-CNN, ktoré k už dvom existujúcim výstupom tejto siete (názvu triedy a bounding-box offsetu) pridáva tretí: objektovú masku. Týmto nám umožňuje s veľkou presnosťou extrahovať podrobnú masku ohraničujúcu nájdený objekt.



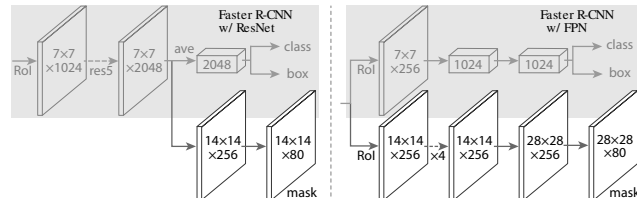
Obr. 2.6: Princíp fungovania konvolučnej neurónovej siete Mask R-CNN. Po prvom kroku RPN (Region Proposal Network), ktorý je zvýraznený modrou farbou nasleduje paralelná časť: klasifikácia tried (horná časť s nápisom „class box“) a určenie masky pokrývajúcej objekt (spodná časť pozostávajúca z dvoch vrstiev)<sup>6</sup>.

Presnejšie Mask R-CNN používa rovnakú dvojkrovú štruktúru ako Faster R-CNN. Prvý krok, RPN (Region Proposal Network), je identický u oboch neurónových sietí. Avšak v druhom kroku je pridané generovanie binárnej masky prebiehajúce paralelne s klasifiká-

<sup>5</sup>Obrázok bol preložený do slovenčiny a pochádza zo štúdie [15]

<sup>6</sup>Obrázok pochádza zo štúdie [11]

ciou triedy a ohraničujúceho obdĺžnika. Výskumníci v tejto štúdií taktiež experimentovali s prepojením určovania triedy a generovaním masky tak, že by na sebe navzájom záviseli. S týmto prístupom experimentovali už štúdie pred nimi, ktoré rovnako skúmali riešenia extrahovania masky objektu. Avšak po vytvorení tejto závislosti generovanej masky na určenej triede, neurónová sieť dosahovala výrazne horšie výsledky. Preto sú tieto dve vetvy vykonávané paralelne. Túto paralelnosť výpočtov masky a klasifikácie možno vidieť na obrázku 2.7



Obr. 2.7: Vizualizácia rozšírenia existujúcej Faster R-CNN metódou Mask R-CNN. Možno vidieť rozšírenie dvoch Faster R-CNN „hláv“. Na obrázku je presne viditeľná paralelnosť výpočtu masky objektu<sup>7</sup>.

Mask R-CNN navrhuje a implementuje nový spôsob extrakcie čírt, RoIAlign, pre presnejšie generovanie výslednej masky. Slúži ako náhrada za bežne používaný RoIPool. Jedinou zmenou oproti metóde RoIPool je nahradenie kvantovania hraníc ohraničujúceho obdĺžniku bilineárnou interpoláciou. Nepresnosť, ktorá vzniká kvantovaním nepredstavuje problém pri klasifikácii objektov a práve preto sieť Faster R-CNN využíva metódu RoIPool. Avšak pri generovaní masky, nahradenie kvantovania a odstránenie nepresností s ním spojených, zapríčinilo výrazné zlepšenie. Konkrétne bolo zaznamenané zlepšenie o 10% v priemernej presnosti pri použití RoIAlign oproti využitiu RoIPool [11].

Model Mask R-CNN svojimi výsledkami prekonal všetky predchádzajúce popredné modely vrátane MNC a FCIS, ktoré boli víťazmi COCO segmentačnej výzvy v rokoch 2015 a 2016. Pri dosahovaní týchto výsledkov bol čas potrebný na spracovanie jedného snímku 200ms.

## 2.3 Translácia medzi súradnicovým systémom kamery a reálneho sveta

Pre presné určenie polohy detegovanej osoby je potrebné poznať pozíciu kamery v priestore, v ktorej bola osoba zachytená. Určenie tejto presnej polohy spočíva vo vykonaní translácie medzi dvomi súradnicovým systémami: (1) kamery a (2) reálneho sveta (monitorovaného objektu). Tento problém je riešený v mnohých štúdiách, kde je nazývaný ako Problém n-bodovej perspektívy (PnP - Perspective n-point problem). Napríklad štúdia [10] predstavuje metódu riešenia P3P problému (Perspective-three-point problem) alebo novšia štúdia [22] prezentuje efektívnu metódu na riešenie PnP problému. Tieto štúdie sa líšia presnosťou výsledného riešenia, komplexnosťou výpočtov a počtu bodov potrebných na vyriešenie.

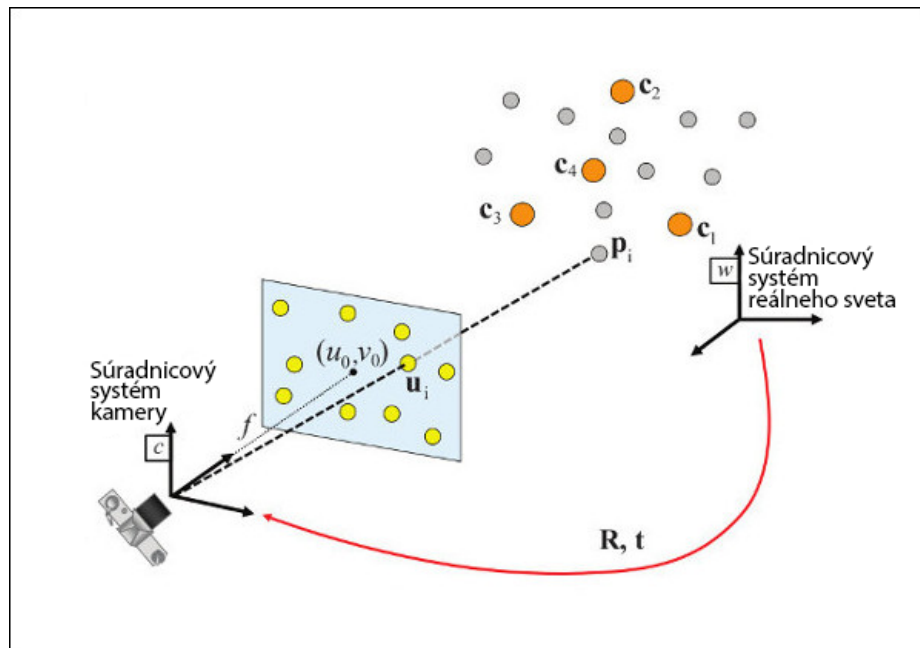
Princíp riešenia tohto problému avšak zostáva rovnaký, je potrebné poznať súradnice bodov v oboch sústavách, kamery aj reálneho sveta. Keďže optika kamery deformuje obraz reálneho sveta, je potrebné počítať s touto deformáciou a kompenzovať ju. Na túto kompenzáciu je nutné poznať parametre použitej optiky.

<sup>7</sup>Obrázok pochádza zo štúdie [11]

Vyššie spomínané štúdie riešia transláciu medzi dvomi súradnicovými systémami inými spôsobmi avšak ich výstup je rovnaký: rotačno-translačná matica, predstavujúca orientáciu kamery voči východiskovému bodu súradnicového systému reálneho sveta. Z tejto matice a z parametrov použitej optiky je možné zostaviť nasledujúcu rovnicu:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Kde  $[X,Y,Z]$  reprezentujú 3D koordináty reálneho sveta,  $f_x$  a  $f_y$  sú ohniskové vzdialenosti optiky,  $c_x$  a  $c_y$  vyjadrujú hlavné body použitej optiky,  $[R|T]$  je rotačno-translačná matica,  $s$  vyjadruje škálu a body  $[u,v]$  predstavujú 2D koordináty v kamerovej projekcii. Tieto hodnoty sú vizualizované na obrázku 2.8.



Obr. 2.8: Vizualizácia PnP.  $R$  a  $t$  predstavuje rotačnú a translačnú maticu. Body  $c_1$  až  $c_4$  sú na-anotované a využité na kalibráciu. Translácia bodov medzi systémami je ilustrovaná bodmi  $u_i$  a  $p_i$ . Bod so súradnicami  $u_0, v_0$  v snímku  $f$  predstavuje neznámy hľadaný bod<sup>8</sup>.

Z tohoto vzťahu je možné prideliť ktorémukoľvek pixlu na obrázku jeho polohu v súradnicovom priestore monitorovaného objektu. Pri použití viacerých kamier s prekrývajúcimi sa zornými uhlami sú získané viaceré detekcie patriace jednej osobe, z každej kamery z ktorej bola detegovaná. Tieto detekcie sa môžu líšiť v póze v akej daná kamera osobu zachytila. Táto zmena pózy môže znamenať aj zmenu ohraničujúceho obdĺžnika detekcie a následne odlišnú polohu, keďže poloha osoby je závislá na tomto obdĺžniku. Združovaniu týchto odlišných polôh patriacej jednej osobe do jednej polohy sa zaoberá štúdia [33]. Zjednotenie rozdielnych bodov nastáva pri splnení dvoch podmienok:

<sup>8</sup>Obrázok bol preložený do slovenčiny a pochádza z dokumentácie knižnice Opencv <https://docs.opencv.org/4.x/pnp.jpg>

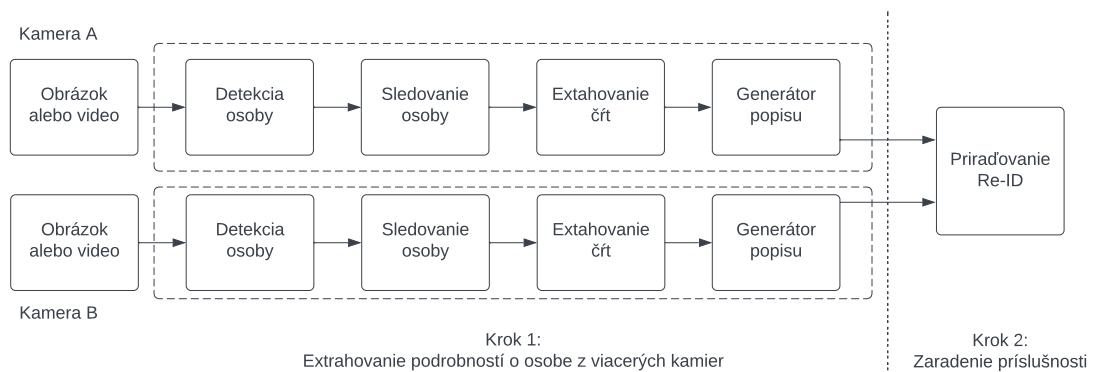
- Zjednotené môžu byť len tie body ktorých Euklidovská vzdialenosť je menšia ako prednastavený prah.
- Zjednotené môžu byť len body z rôznych kamier, keďže sa jedna osoba nemôže na jednej kamere nachádzať dvakrát.

Na samotné zjednocovanie je použitá hladná heuristika (greedy heuristic) bodov zoradených podľa  $x$ -ovej súradnice.

## 2.4 Sledovanie ľudí naprieč snímkami a kamerami

V prostredí múzea, galérie či iného priestoru, v ktorom chceme monitorovať pohyb ľudí, sa budú vyskytovať miesta, v ktorých sa zorné uhly jednotlivých kamier neprekrývajú prípadne vôbec nie sú pokryté kamerami. Pre úspešné monitorovanie osoby naprieč objektom ju musíme po jej detegovaní identifikovať, teda vytvoriť súbor črt špecifických pre danú osobu. Jej následné sledovanie spočíva v opakovanej identifikácii danej osoby naprieč ďalšími snímkami. Tomuto procesu sa vraví re-identifikácia (Re-ID) a je základným aspektom multi-kamerového sledovania. Detegovanie a sledovanie osôb na jednotlivých kamerách spojené s re-identifikáciou nám umožňuje rekonštruovať trajektóriu sledovanej osoby naprieč objektom.

V princípe je táto činnosť rozdelená na získanie črt neznámej osoby a ich následné porovnanie s existujúcim súborom sledovaných osôb [3]. Vyjadrením príslušnosti novo získaných črt k danej osobe je získaná najpravdepodobnejšia zhoda, ktorá identifikuje neznámu osobu. Tento princíp je znázornený na obrázku 2.9.



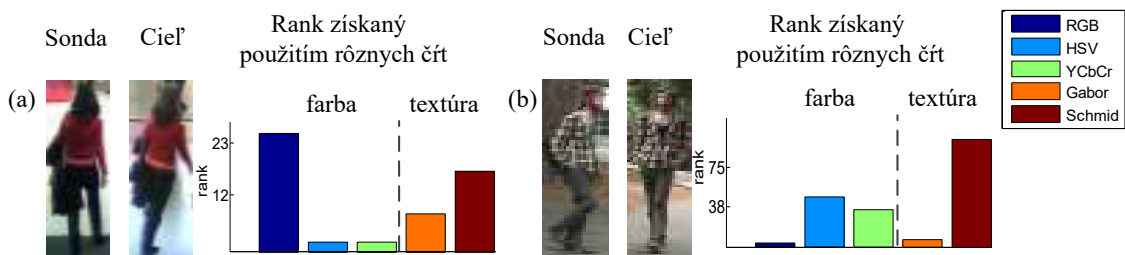
Obr. 2.9: Blokový diagram popisujúci fungovanie Re-ID systému<sup>9</sup>.

Väčšina existujúcich metód extrahovania črt kombinuje rozoznávanie niekoľkých vlastností osoby primárne však typu, farby alebo textúry oblečenia. Tieto metódy sa zameriavajú na tzv. krátkodobú re-identifikáciu (short-period Re-ID) [3], ktorá sa spolieha na nemenosť črt sledovaných osôb v krátkom časovom rozmedzí. Táto krátkodobá re-identifikácia sa nezaobrá sledovaním subjektov naprieč ich viacerými návštevami sledovaného objektu. Sledovaným črtám subjektu môžu byť pri re-identifikácii pridelené rovnaké váhy. Štúdia [24] skúma alternatívny prístup prerozdelenia váh na základe ich dôležitosti a jednoznačnosti.

<sup>9</sup>Obrázok bol preložený do slovenčiny a pochádza zo štúdie [3]



Na obrázku 2.10 môžeme vidieť tento princíp, ktorý by priradil menej jednoznačnej čрте menšiu váhu.



Obr. 2.10: Zobrazenie testovaného (Probe) snímku s referenčným (Target) snímkom a miera ich zhody na základe rôznych črt<sup>10</sup>.

Na presné určenie farby oblečenia detegovanej osoby je vhodné použiť histogram farieb. Tento histogram zobrazuje frekvenciu v akej sa jednotlivé farby nachádzajú v snímku. Osa X reprezentuje rozsah intenzity farby a osa Y zase početnosť danej intenzity. Histogram farieb s rozsahom intenzity farieb  $[0, L - 1]$  sa dá zapísať ako funkcia

$$h(r_k) = n_k$$

kde  $r_k$  predstavuje  $k$ -tu intenzitu farieb a  $n_k$  je počet pixlov, ktorých intenzita danej farby je rovná  $r_k$  [19]. V prípade využitia RGB modelu reprezentácie farieb, kde sú všetky zobrazené farby vyjadrené ako kombinácia troch základných farieb (červenej - R, zelenej - G a modrej - B) sa osa X najčastejšie uvádza v rozmedzí 0 až 255. Kde 0 vyjadruje najmenšiu intenzitu farieb a 255 najvyššiu. Takýto histogram farieb možno vidieť na obrázku 2.11.

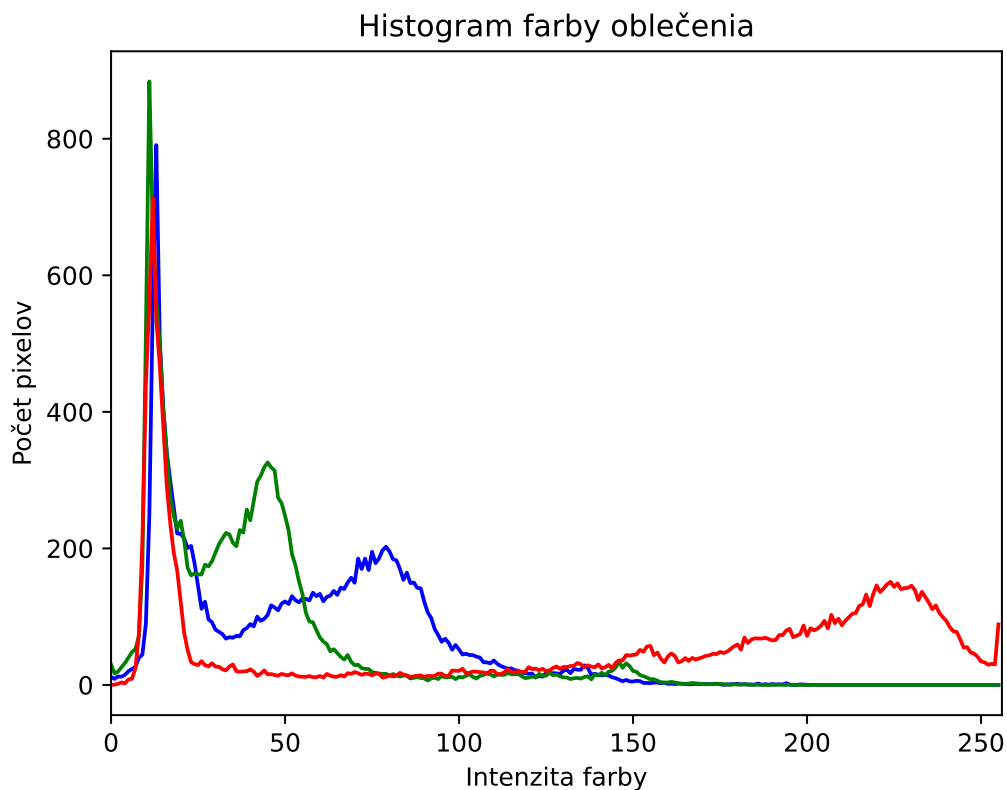
Na zistenie podobnosti dvoch histogram možno využiť Hellingerovu vzdialenosť. Táto vzdialenosť sa používa na vyjadrenie podobnosti medzi dvomi rozdeleniami pravdepodobnosti.

## 2.5 Priradenie detekcií existujúcim osobám

Na extrahovanie črt osoby, popísané v predchádzajúcej kapitole, naväzuje problém čo najideálnejšieho priradenia nových detekcií k už existujúcim osobám. Jedná sa o stanovenie vzájomnej podobnosti a efektívnom priradení najviac podobných prvkov k sebe. Vzájomnú podobnosť medzi novými detekciami a existujúcimi osobami možno označiť ako *cenu* daného priradenia. Funkciu ceny tohoto priradenia je možno vyjadriť ako vzdialenosť (rozdielnosť) súboru črt detekcie a súboru črt existujúcej osoby. Priamočiary prístup k tomto priradeniu by bolo vytvorenie permutácie všetkých detekcií a všetkých existujúcich osôb. Následné by sa vypočítala cena každého z týchto priradení a bola by vybraná možnosť s najmenšou cenou. Tento jednoduchý prístup by avšak bol časovo neprijateľný v ľuďmi veľmi frekventovaných miestach, keďže by zvýšenie počtu ľudí či detekcií viedlo k exponenciálnemu zvýšeniu počtu permutácií.

Iný pohľad na túto problematiku by poskytlo vnímanie existujúcich osôb a nových detekcií ako dvoch skupín s určitými vzájomnými vzťahmi. Podľa obmedzení a vlastností týchto vzťahov možno vykonať transformáciu tohto problému priradenia na dva rozdielne

<sup>10</sup>Obrázok bol preložený do slovenčiny a pochádza zo štúdie [24]



Obr. 2.11: Histogram farieb osoby, ktorá má na sebe oblečené tmavé nohavice a červenú bundu. Osa x znázorňuje farebnú intenzitu farby a osa y počet pixlov na obrázku, ktoré majú príslušnú intenzitu farby. Možno si všimnúť výrazne vysoké hodnoty v dolnej časti intenzity, predstavujúce tmavú farbu. Rovnako vo vyšších intenzitách dominuje práve červená farba.

problémov. Prvý, predstavuje transformáciu na párný graf (kapitola 2.5.1). Druhá možnosť reprezentuje riešenie za pomoci Munkresovho algoritmu, tiež známeho ako maďarský algoritmus (kapitola 2.5.2).

### 2.5.1 Párny graf

Párnym grafom je označovaný taký graf, ktorého množinu vrcholov možno rozdeliť na dve disjunktné množiny tak, že žiadne dva vrcholy pochádzajúce z tej istej množiny nie sú spojené hranou. Graf  $G(V, E)$  s dvomi množinami vrcholov  $X$  a  $Y$ , označované ako *strany* párneho grafu, je párnym grafom ak platí nasledovné [32]:

$$\begin{aligned}
 V &= X \cup Y \\
 X \cap Y &= \emptyset \\
 \forall e = \{u, v\}, e \in E : u \in X \wedge v \in Y
 \end{aligned}$$

Orientovaný párný graf vyžaduje, aby boli všetky hrany orientované súhlasne. Teda platí  $E(G) = W^+(X)$  kde  $W^+(X)$  značí množinu všetkých hrán, ktorých počiatočný vrchol leží v  $X$  a koncový vrchol neleží v  $X$  [7].

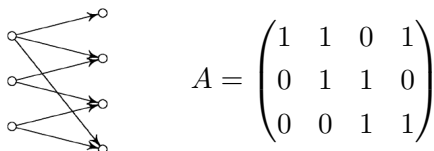
Váňovaný párnym grafom je označovaný taký graf  $G(X, Y, W)$  s  $W = (w_{ij})$  kde  $w_{ij} > 0$  značí váhu hrany medzi vrcholmi  $i$  a  $j$ . Ak sa  $w_{ij} = 0$  znamená to, že medzi vrcholmi  $i$  a  $j$  hrana neexistuje [32].

Taktiež existuje *úplný párný graf*, v ktorom každá dvojica vrcholov  $x \in X$  a  $y \in Y$  je spojená presne jednou hranou. Takýto graf je označený ako  $K_{m,n}$  kde  $m$  značí veľkosť množiny  $X$  a  $n$  veľkosť množiny  $Y$ . Platí že takýto graf má  $mn$  hrán.

Matica susednosti párneho grafu popisuje, ktoré vrcholy v grafe sú spojené hranou. Je založená na skutočnosti, že vrcholy párneho grafu sa dajú usporiadať tak, aby matica susednosti mala tvar

$$M_G^+ = \begin{pmatrix} 0 & A \\ 0 & 0 \end{pmatrix} \text{ alebo } M_G = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix}$$

v závislosti na tom či sa jedná o párný graf orientovaný ( $M_G^+$ ) alebo neorientovaný ( $M_G$ ). Podmaticu  $A$  potom nazývame maticou susednosti párneho grafu. Príklad tejto matice je vidieť na obrázku 2.12. V prípade váňovaného párneho grafu sa budú v matici susednosti nachádzať váhy jednotlivých hrán grafu.



Obr. 2.12: Orientovaný párný graf s príslušnou maticou susednosti<sup>11</sup>.

Na problém priradenia osôb detekciám možno využiť riešenia fungujúce pre párne grafy. Za množinou  $X$  si možno predstaviť novo nájdené detekcie a za množinou  $Y$  už existujúce osoby. Hrana medzi dvoma vrcholmi grafu pochádzajúcich z rozdielnych množín je vytvorená, ak novo nájdená detekcia spĺňa podmienky priradenia k existujúcej osobe. Váha tejto hrany je určená na základe podobnosti novo nájdenej detekcie s existujúcou osobou podľa črt, ktoré popisuje kapitola 2.4. Na rozhodnutie priradenia detekcií osobám zostáva po vytvorení párneho grafu vyriešiť problém párovania tohto grafu. Podľa toho, či sa jedná o úplný alebo neúplný párný graf možno využiť rozdielne metódy na vyriešenie problému párovania grafu.

V prípade párovania detekcií osobám možno predpokladať, že nie každá detekcia bude môcť byť kandidátom pre každú osobu a tým pádom sa nebude jednať o úplný párný graf. Mimo diametrálne rozličných črt, ktoré zamedzia kandidovaniu detekcie daným osobám a tým zamedzeniu splnenia podmienky úplnosti párneho grafu sa môžu vyskytnúť prípady, kedy do monitorovanej oblasti vstúpi nová osoba a nesplní podmienky priradenia k žiadnej osobe. V tomto prípade nebude existovať hrana spájajúca túto detekciu so žiadnou osobou. Z tohoto dôvodu možno zvažovať riešenia, ktoré sa zaoberajú práve neúplnými párnymi grafmi. Jedno z týchto riešení je aj Hopcroft–Karpov algoritmus [13]. Jeho implementácia v knižnici SciPy<sup>12</sup> dosahuje časovú komplexnosť  $O(|E|\sqrt{|V|})$  pričom jeho priestorová náročnosť rastie lineárne s počtom riadkov v matici susednosti.

<sup>11</sup>Obrázok pochádza z knihy [7]

<sup>12</sup><https://scipy.org/>

## 2.5.2 Munkresov algoritmus

Munkresov algoritmus, tiež známy ako maďarský algoritmus, je úpravou už existujúceho algoritmu Harolda Kuhna Jamesom Munkresom v roku 1957 [26]. Algoritmus rieši problém priradenia v polynomiálnom čase.

Problém priradenia predstavuje základný optimalizačný problém. Môžeme ho vyjadriť ako vytvorenie vzťahu medzi prvkami dvoch množín tak, že cena ich priradenia bude minimálna. Pri tomto priradení musí platiť, že ktorýkoľvek prvok z prvej množiny môže byť priradený ktorémukoľvek prvku z druhej množiny. Rovnako platí, že prvok z ktorejkoľvek množiny môže byť priradený len jednému prvku z ďalšej množiny. Uvažujme napríklad určitý počet pracovníkov a určitý počet úloh ako dve množiny pre priradenie. Ktorýkoľvek pracovník môže byť priradený ktorejkoľvek úlohe, pričom každé priradenie má určitú cenu závislú od kombinácie pracovníka a úlohy. Každý úlohe môže byť priradený maximálne jeden pracovník pričom pracovník môže mať priradenú maximálne jednu úlohu. Priradovanie sa uskutočňuje tak, aby bol súčet cien všetkých priradení minimálny. Ak sa počet pracovníkov a úloh rovnajú, jedná sa o *vybalancované priradenie* inak o *nevybalancované priradenie*.

Tento problém, ako ho popisuje štúdia [26] a [28] možno matematicky zdefinovať nasledovne. Existuje  $C_{ij}$ ,  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$ , ktoré predstavuje cenu vykonania j-tej úlohy i-tým pracovníkom. Ďalej je možno označiť indikátor priradenia  $X_{ij}$  tak, že platí:

$$X_{ij} = \begin{cases} 1 & \text{ak } i\text{-ty pracovník má priradenú } j\text{-tu úlohu} \\ 0 & \text{ak } i\text{-ty pracovník nemá priradenú } j\text{-tu úlohu} \end{cases}$$

Cieľom Munkresovho algoritmu je minimalizovať cenu priradenia  $Z$  vypočítanú ako:

$$Z = \sum_{i=1}^m \sum_{j=1}^n C_{ij} X_{ij}$$

Zároveň je pri riešení nutné dodržať podmienku maximálneho priradenia jednej úlohy jednému pracovníkovi vyjadrenú ako:

$$\sum_{i=1}^m X_{ij} = 1, j = 1, 2, \dots, n$$

$$\sum_{j=1}^n X_{ij} = 1, i = 1, 2, \dots, m$$

$$X_{ij} = 0 \vee 1$$

Riešenie munkresovho algoritmu spočíva vo 4 krokoch [17]. Na začiatok je nutné zostaviť maticu cien jednotlivých priradení, podobajúcej sa do určitej miery matici susednosti z kapitoly 2.5.1. Napríklad nasledovnú maticu, v ktorej stĺpce predstavujú jednotlivé úlohy, riadky pracovníkov a hodnoty v matici ceny vykonania danej úlohy:

$$\begin{pmatrix} 30 & 25 & 10 \\ 15 & 10 & 20 \\ 25 & 20 & 15 \end{pmatrix}$$

**Krok 1** využíva kľúčovú vlastnosť, ktorá je základom maďarského algoritmu. Riešenie sa nezmení ak je od všetkých hodnôt v riadku či stĺpci odpočítaná tá istá hodnota. Za túto

hodnotu sa vyberie najmenšia hodnota v riadku, ktorá sa následne odpočíta sa od všetkých hodnôt v danom riadku. Najmenšia hodnota sa vyberá práve preto aby sa po vykonaní tejto operácie v každom riadku matice nachádzal práve jeden prvok s nulovou hodnotou:

$$\begin{pmatrix} 20 & 15 & 0 \\ 5 & 0 & 10 \\ 10 & 5 & 0 \end{pmatrix}$$

Následne sa vykoná kontrola na ideálne priradenie, ktorá vyžaduje, aby sa v každom stĺpci nachádzala práve jedna nula. Ak je táto podmienka splnená, úloha sa prideli tomu pracovníkovi, ktorý má pri danej úlohe nulovú hodnotu. Ak táto podmienka nie je splnená, je nutné pokračovať na ďalší krok.

**Krok 2** je veľmi podobný prvému kroku. Narozdiel od hľadania najmenšieho prvku v riadku je hľadaný najmenší prvok v stĺpci a následne odpočítaný od celého stĺpca:

$$\begin{pmatrix} 15 & 15 & 0 \\ 0 & 0 & 10 \\ 5 & 5 & 0 \end{pmatrix}$$

**Krok 3** spočíva v spojení všetkých nulových prvkov v matici čo najmenším počtom priamok.

$$\begin{pmatrix} 15 & 15 & 0 \\ 0 & 0 & 10 \\ 5 & 5 & 0 \end{pmatrix}$$

Ak sa počet priamok rovná počtu stĺpcov, bolo nájdené optimálne riešenie. Ak táto rovnosť nenastala je nutné pokračovať ďalším krokom.

**Krok 4** predstavuje posledný krok Mukresovho algoritmu. Z matice z predchádzajúceho kroku sú vybrané elementy, ktoré neležia pod žiadnou z vytvorených priamok. Z týchto prvkov je nájdený ten najmenší, ktorého hodnota je následne odpočítaná od všetkých prvkov, ktoré neležia pod priamkami a pripočítaná ku všetkým prvkom, ktoré ležia pod priesečníkom dvoch priamok.

$$\begin{pmatrix} 10 & 10 & 0 \\ 0 & 0 & 15 \\ 0 & 0 & 0 \end{pmatrix}$$

Ak stále nie je možné priradenie opakujeme kroky 4 a 5 pokým priradenie bude možné. V tomto príklade je priradenie možné dokonca sú na výber dve možnosti.

$$\begin{pmatrix} 10 & 10 & 0 \\ 0 & 0 & 15 \\ 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 10 & 10 & 0 \\ 0 & 0 & 15 \\ 0 & 0 & 0 \end{pmatrix}$$

Pre zistenie výslednej ceny priradenia sú vybrané hodnoty na daných miestach v pôvodnej matici. Na zistenie celkovej ceny priradenia sú tieto hodnoty sčítané. V tomto príklade je jedno, ktorý z výsledkov je zvolený, keďže obe možnosti majú rovnakú výslednú cenu priradenia a to 45.

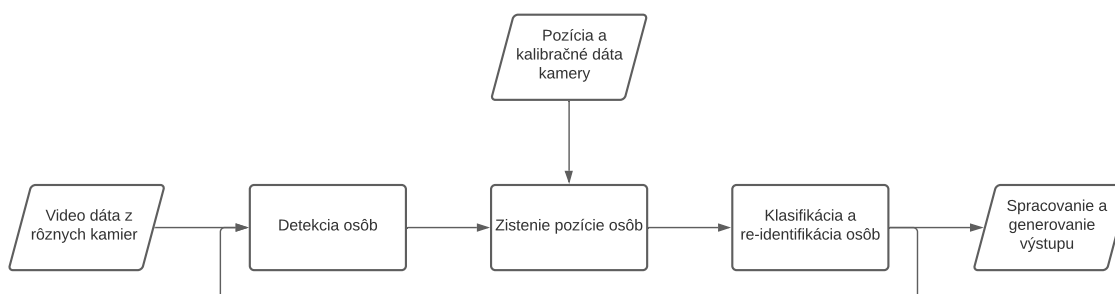
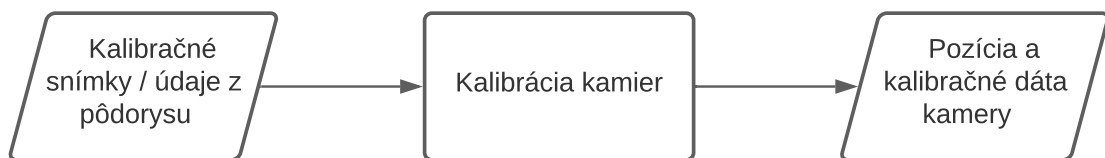
$$\begin{pmatrix} 30 & 25 & 10 \\ 15 & 10 & 20 \\ 25 & 20 & 15 \end{pmatrix} \quad \begin{pmatrix} 30 & 25 & 10 \\ 15 & 10 & 20 \\ 25 & 20 & 15 \end{pmatrix}$$

Tento algoritmus je možné využiť na problém pridelovania detekcií osobám, ak sú za pracovníkov dosadené nové detekcie, za úlohy existujúce osoby a ako cenová funkcia je zvolená podobnosť medzi detekciou a osobou. Pri vytváraní matice cien neexistuje možnosť, ktorú by poskytovala priamo definícia Mukresovho algoritmu, ktorá by umožňovala zabráneniu priradeniu detekcie osobe inak, ako neuvedením danej detekcie v matici. V prípade, že neexistuje dostačujúca podobnosť medzi danou detekciou a osobou je nutné zabrániť takémuto priradeniu. Preto je vhodné vybrať si vhodné, dostatočne veľké číslo, ktoré je väčšie než všetky ceny priradenia. Toto číslo bude vložené do matice na miesta dvojíc medzi ktorými neexistuje hrana a teda je zakázané priradenie. Tento prístup by mal odradiť algoritmus od vyberania párov detekcia-osoba, ktorých cena je toto vhodne vybrané číslo. Táto skutočnosť sa dá následne overiť spätnou kontrolou ceny priradenia a vylúčením priradení, ktorých cena je toto vhodne vybrané číslo.

# Kapitola 3

## Návrh

Návrh systému je logicky členený na dva celky, inicializácia a kalibrácia kamier a samotné spracovávanie obrazových dát. Sú názorne ilustrované na obrázku 3.1 a bližšie popísané v tejto kapitole.



Obr. 3.1: Inicializačný 3.1a a funkčný 3.1b blokový diagram ilustrujúci navrhovaný spôsob fungovania systému. Zobrazuje jednotlivé moduly systému a znázorňuje tok informácií systémom.

### 3.1 Kalibrácia kamier - inicializácia

Prvým krokom pri sledovaní návštevníkov je kalibrácia kamier. Je potrebné zistiť pozíciu a natočenie kamery v rámci súradnicového systému reálneho sveta, aby bolo možné získať presnú polohu návštevníkov. Tento problém je vyriešiteľný za použitia metódy opísanej v kapitole 2.3. Pre jeho vyriešenie je nutné zistiť polohy kľúčových bodov a parametre optiky s ktorou pracujeme.

Prvým krokom je identifikácia kľúčových bodov, ktorým vieme priradiť súradnice a sú jednoznačne rozoznateľné na snímkach. Takýmito bodmi môžu byť rohy dverí, vitrín s expo-

zíciami či spoločné rohy stien. Následne sú z tejto množiny kľúčových bodov vybrané tie, ktoré sú v zornom uhle kamery a sú im priradené ich pozície v súradnicovom priestore kamery (súradnica pixlu s kľúčovým bodom). Presné pozície týchto vhodných bodov, v súradnicovom priestore monitorového objektu, je možné zistiť z pôdorysu monitorového objektu. Pri absencii pôdorysu je možné potrebné súradnice zistiť meraním priamo na mieste. Pri využívaní pohľadov z viacerých kamier umiestnených vo viacerých miestnostiach je nutné zachovať rovnaký počiatok súradnicovej sústavy pre všetky body.

Druhú časť údajov potrebných na tento výpočet predstavujú vnútorné parametre kamery ako sú ohniskové vzdialenosti jej objektívu a hodnotu hlavného bodu. Tieto údaje môžu byť zovšeobecnené na základné hodnoty  $f_x = f_y =$  šírka snímku a  $c_x =$  šírka snímku/2  $c_y =$  výška snímku/2, ktoré zodpovedajú väčšine bežných objektívov. Avšak pre špeciálne typy objektívov, ako sú napríklad širokouhlé objektívy či naopak objektívy s úzkym zorným uhlom, treba tieto údaje dôkladne nakalibrovať. Dajú sa vyčítať z technických parametrov danej kamery/objektívu alebo externe experimentálne vypočítať.

Pozícia kamery v súradnicovom priestore monitorovaného objektu bude následne uložená a použitá pri samotnom monitorovaní návštevníkov. V prípade potreby môže byť znovu, v pravidelných intervaloch počítaná.

## 3.2 Detekcia a re-identifikácia osôb

Ako je možno vidieť z obrázku 3.1b, samotný systém bude pozostávať z niekoľkých častí. Pri návrhu bol kladený dôraz na flexibilitu systému. Môžeme predpokladať, že sa budú vyskytovať veľmi špecifické požiadavky na fungovanie systému naprieč rôznymi objektmi, v ktorých bude nasadený. Napríklad môžeme uvažovať, že pri skúšobnej fáze systému bude k dispozícii pohľad len z jednej kamery alebo systém nebude mať prístup k dátam v reálnom čase a bude tým pádom potrebné pracovať len s už vyhotovenými nahrávkami. Aby bolo možné systém týmto požiadavkám prispôbovať bol navrhnutý tak, aby všetky časti pracovali do určitej miery samostatne.

Pre každú časť systému existuje abstraktná trieda, ktorej funkcie musí každá časť implementovať. Formát vstupu aj výstupu dát z týchto funkcií je v rámci systému „štandardizovaný“. Systém ako celok sa spolieha len na funkcie definované v týchto abstraktných triedach pričom nie je zafazovaný spôsobom, akým sú dáta v jednotlivých častiach spracovávané. Napríklad rozdiel medzi dvomi systémami spracovávajúcimi záznam (1) z jednej kamery a (2) z niekoľkých kamier je len v jedinej triede, triede načítania obrazových dát. Zostatok systému túto zmenu nepocítiť.

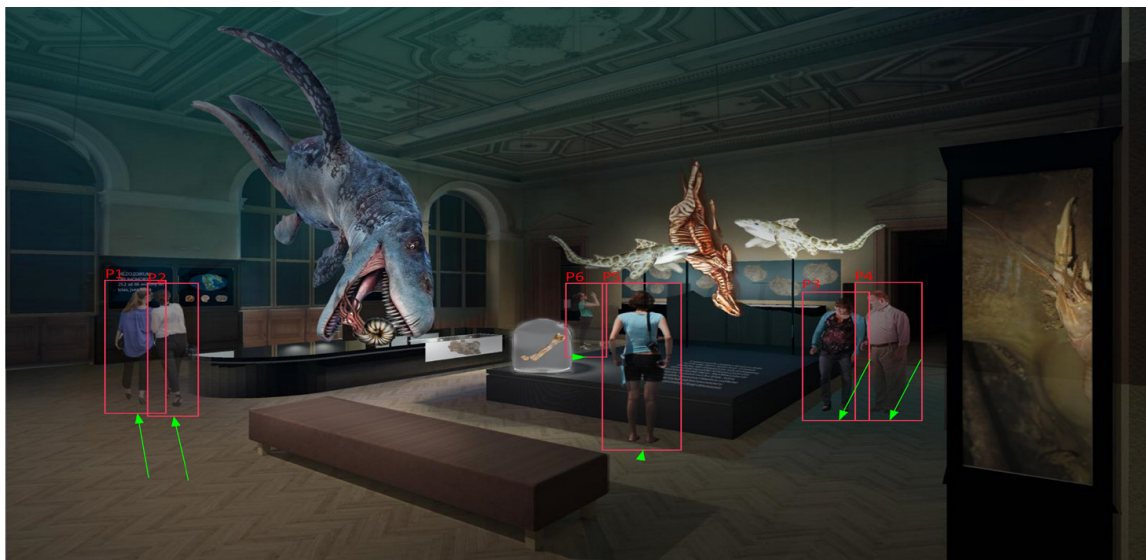
Tento kompartmentovaný prístup k systému taktiež umožňuje jeho jednoduchú aktualizáciu. Pri objavení nových metód identifikácie alebo nových konvolučných sietí na detegovanie ľudí stačí nahradiť len príslušné časti systému.

Navrhovaný pipeline systému vyzerá nasledovne. Modul spracovávajúci vstup načíta snímky z preddefinovaných zdrojov. Tie sú následne poslané modulu na detekciu. V tomto module som sa rozhodol použiť konvolučnú neurónovú sieť Mask R-CNN bližšie rozoberanú v kapitole 2.2.4. Táto sieť vracia okrem ohraničujúcich obdĺžnikov jednotlivých detekcií taktiež masku, pod ktorou sa daný objekt nachádza. Vďaka tejto maske sú z detekcie presne extrahované jej črty, informácie o farbe či textúre. Detegované objekty sú následne odovzdané modulu na zistenie presnej pozície. Na základe konfigurácie kamery popísanej v predchádzajúcej sekcii sa tento bod pretransformuje do súradnicového systému sledovaného objektu, čím sa zistí pozícia osoby v objekte. Bod, ktorý predstavuje pozíciu osoby, je znázornený na obrázku 3.2 vo forme zelenej šípky. Detekcie s ich pozíciami sú predané



modulu na identifikáciu a sledovanie osôb. Tento modul si uchováva informácie o všetkých detekciách, ich pozíciach a čase jednotlivých detekcií. Priradovanie identifikácií postupuje podľa princípov popísaných v sekcii 2.5. Poslednou časťou systému je generovanie výstupu. Z dát z predchádzajúceho modulu zostavuje vizualizáciu pohybu osôb naprieč monitorovaným priestorom. Táto vizualizácia môže predstavovať tepelnú mapu (heatmap) zobrazujúcu miesta, na ktorých návštevníci strávili najviac času alebo zobrazenie trajektórií jednotlivých návštevníkov. Rovnako ponúka hodnotenie prednakonfigurovaných exponátov podľa metrick popísaných v 2.1. Výstup modulu môže byť požadovaný v reálnom čase za každým načítaním nového snímku, na vyžiadanie používateľa alebo po dokončení spracovávania všetkých obrazových záznamov.

Systém možno spúšťať s užívateľským rozhraním alebo bez neho pre najrýchlejšie získanie výstupných dát. Užívateľské rozhranie poskytuje užívateľovi pohľad na aktuálne spracovávaný obrazový záznam, možnosť vyžiadať si výstupné dáta z aktuálne nazbieraných dát ale taktiež zisťovanie podrobností o aktuálne spracovávaných dátach a je bližšie popísané v kapitole 4.5.



Obr. 3.2: Vizualizácia sledovania osôb v múzeu. Zelené šípky na obrázku ukazujú na body, ktorých súradnice sú považované za pozície daných ľudí v priestore<sup>1</sup>.

<sup>1</sup>Obrázok pochádza zo štúdie [8]. Autor: Ing.arch.D.Kříž

## Kapitola 4

# Implementácia

System je implementovaný v jazyku Python<sup>1</sup>. Využíva predovšetkým špeciálne skompilovanú knižnicu Opencv<sup>2</sup> s podporou akcelerácie výpočtov na grafickej karte. Rovnako je tiež v značnej miere využívaná knižnica NumPy<sup>3</sup> a SciPy<sup>4</sup>. Implementácia riadiacej logiky systému sa nachádza v `application.py`. Moduly systému sú ukladané podľa nasledujúcej šablóny: `modules/{názov modulu}/{implementácia modulu}`. Správanie celého systému a jeho modulov je riadené z konfiguračných súborov, ktoré majú koncovku `.config`. Hodnoty v týchto súboroch určujú aké moduly, prahové hodnoty či cesty k vstupným súborom sa použijú. Kalibrácia kamier je implementovaná v súbore `configureCamera.py`. Jej výstup je potrebný pre správne priradenie polohy objektom.

### 4.1 Implementácia kalibrácie kamier

Kalibrácia kamier využíva princípy popísané v kapitole 2.3. Dokáže spracovávať ako samostatné snímky, tak aj celé video. V prípade spracovania celého videa užívateľ má plnú kontrolu nad prehrávaním videa a vybraním vhodnej snímky, na ktorú budú aplikované kalibračné metódy. Sú implementované dve metódy kalibrácie kamery:

- Kalibračná šachovnica. Pozostáva zo šachovnice a bieleho okraja okolo nej. Šachovnica sa skladá z vopred známeho počtu štvorcov so známou veľkosťou strany. Hrúbka bieleho okraja okolo šachovnice by mala mať veľkosť minimálne jednej strany štvorca šachovnice. Absencia tohto okraja môže viesť k nemožnosti nájdenia šachovnice v zázname. Ľavý dolný roh šachovnice je pokladaný za počiatok súradnicovej sústavy reálneho sveta. Požiadavky na šachovnicu pochádzajú z implementácie funkcie `findChessboardCorners()` v knižnici Opencv. Pre vykonanie konfigurácie je potrebné rozoznať šachovnicu spĺňajúcu tieto požiadavky v kamerovom zázname.
- Ručne zvolené oamotované body. Tento prístup si vyžaduje poznať súradnice kľúčových bodov v súradnicovom systéme reálneho sveta ako aj systéme kamery. Na zistenie súradníc v systéme kamery je implementovaná funkcia pri stlačení ľavého tlačidla myši. Súradnice bodov v reálnom svete treba zistiť z pôdorysu budovy alebo z externého merania. Na riešenie problému n-bodovej perspektívy potrebujeme minimálne 4 body.

---

<sup>1</sup><https://www.python.org/>

<sup>2</sup><https://opencv.org/>

<sup>3</sup><https://numpy.org/>

<sup>4</sup><https://scipy.org/>

Avšak platí, že čím viac bodov konfigurácii poskytneme, tým presnejší výsledok môžeme očakávať. Rovnako z testovania vyplynulo, že najlepšie výsledky konfigurácie sa dosahujú pri použití bodov, ktoré ležia aspoň v troch na seba kolmých rovinách, pričom v každej z rovín sa nachádzajú aspoň tri body. Ako príklad je vhodné použiť „kalibračnú krabicu“, ktorú umiestnime na snímku tak, aby boli viditeľné jej 3 strany, pričom ako kalibračné body použijeme jej rohy. „Kalibračnú krabicu“ je vhodné použiť pre zvýšenie počtu dostupných bodov, pre zlepšenie presnosti kalibrácie alebo pri absencii bodov vo všetkých troch rovinách.

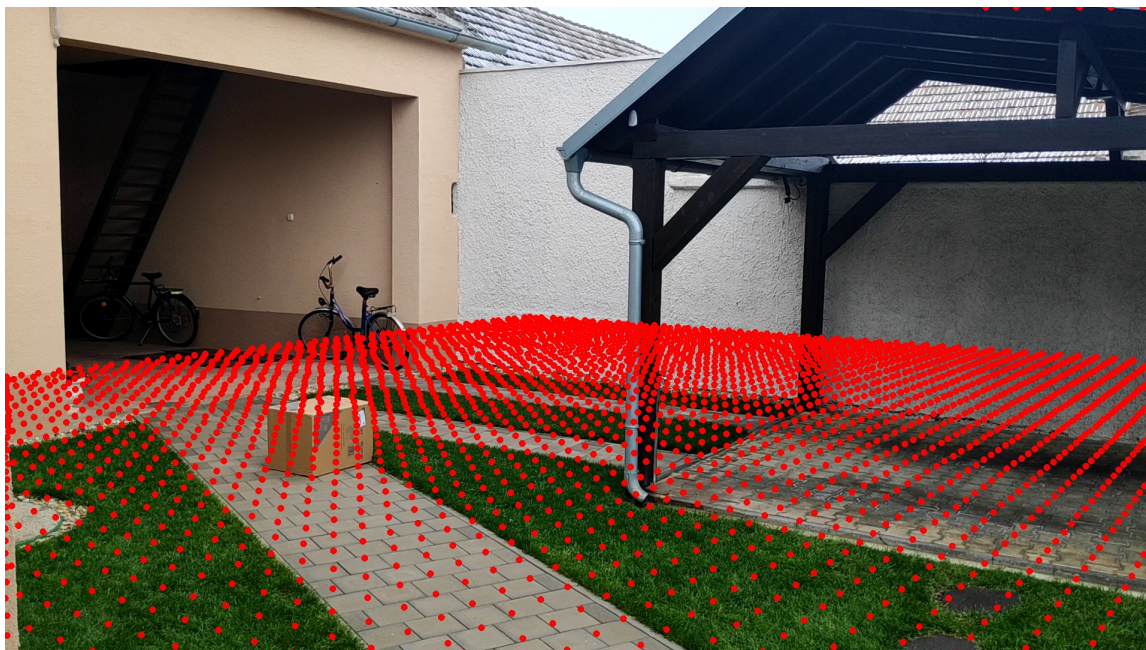


Obr. 4.1: Ručne zvolených 11 bodov použitých na kalibráciu kamery. Sú použité body identifikované z pôdorysu monitorovanej oblasti (vstup do budovy) a body v priestore (rohy vloženej krabice). „Kalibračná krabica“ bola použitá z dôvodu absencie bodov v priestore monitorovanej oblasti a pre zlepšenie presnosti konfigurácie.

Z vlastného testovania som prišiel k záveru, že najspoľahlivejšie funguje druhá metóda, ručne zvolených oantovaných bodov. Metóda kalibračnej šachovnice vyžaduje dobrú viditeľnosť šachovnice, jej správne natočenie a čo najmenšiu vzdialenosť ku kamere. Preto je metóda spoliehajúca sa na ručne oantované body flexibilnejšia a vcelku vhodnejšia všade, kde sa nachádza dostatok týchto kľúčových bodov, keďže nepotrebuje v zázname detegovať šachovnicu. Metódu kalibračnej šachovnice je aj napriek tomu vhodné využiť v špeciálnych podmienkach, zjavne v malých interiérových priestoroch bez výrazných kľúčových bodov, v ktorých nie je možné použiť metódu ručne zvolených oantovaných bodov.

Pri využití ktoréhokoľvek z vyššie spomínaných prístupov treba taktiež poskytnúť vnútorné parametre použitej optiky alebo použiť východiskové parametre ako sú popísané v sekcii 3.1.

Na overenie správnosti vykonanej kalibrácie sú implementované dve funkcie. Prvá, zachytáva kliknutia pravého tlačítka myši a vypíše pozíciu kliknutého bodu v oboch súradnicových priestoroch. Druhá, vykreslí v 20 cm intervaloch rovinu predstavujúcu zem. Týmto



Obr. 4.2: Vizualizácia roviny predstavujúcej zem. Na zistenie polohy kamery bolo využitých 11 bodov so známymi súradnicami (rohy vstupu do budovy + rohy krabice v priestore).

spôsobom sa dá ľahko vizuálne zistiť naklonenie tejto roviny a teda aj nesprávnosť konfigurácie. Tento spôsob vykreslenia je možno vidieť na obrázku 4.2.

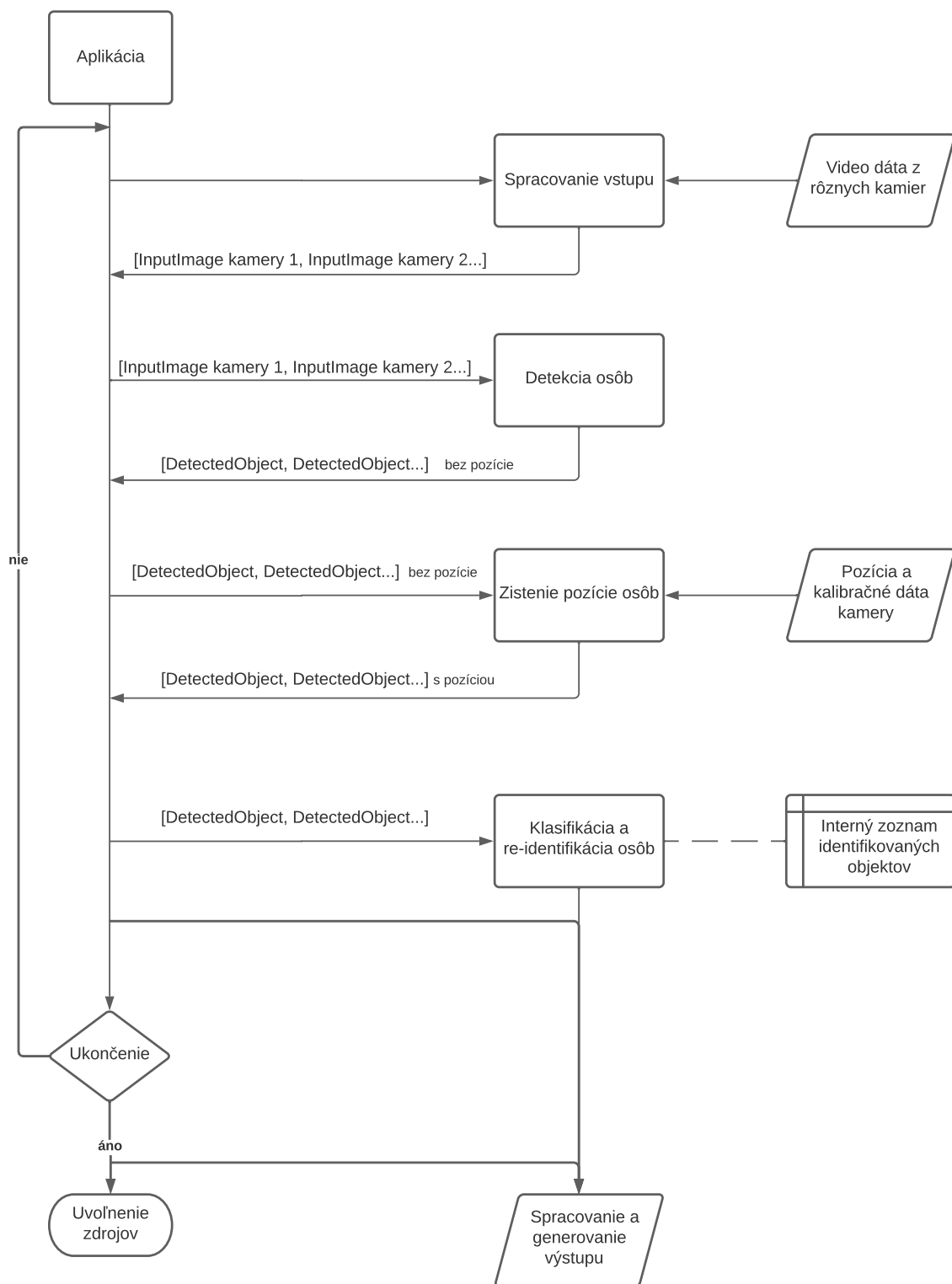
Výsledná konfiguračná matica je uložená do priečinku, v ktorom sa nachádza konfiguračný skript. Je uložená vo formáte pickle (koncovka `.pkl`).

## 4.2 Implementácia riadiaceho systému

Na začiatku tejto kapitoly bol navrhovaný systém predstavovaný ako súbor spolupracujúcich modulov. Komunikácia medzi týmito modulmi je znázornená na obrázku 4.3.

Samotné jadro systému, `application.py`, je zodpovedné za načítavanie jednotlivých modulov a odovzdávanie informácií medzi nimi. Moduly nekomunikujú medzi sebou ale všetky údaje dostávajú z jadra systému a spracované mu ich naspäť odovzdávajú. Tento spôsob komunikácie má za následok väčšiu nezávislosť jednotlivých modulov, ich ľahšiu vymeniteľnosť a vytvára jeden centrálny bod riadenia chodu systému. Keďže centrálny bod má prístup ku všetkým informáciám, je v ňom implementované aj zobrazovanie užívateľského rozhrania. Užívateľské rozhranie predstavuje funkcie na riadenie chodu systému, zobrazovanie dát a prácu s týmito dátami a je bližšie popísané v kapitole 4.5.

Každý z modulov musí minimálne implementovať rozhranie, ktoré je definované v abstraktnej triede patriacej príslušnému modulu. V nasledujúcich odstavcoch budú popísané jednoduchšie moduly systému, pričom zložitejšie moduly budú bližšie vysvetlené v kapitolách 4.3 a 4.4.



Obr. 4.3: Diagram komunikácie znázorňujúci spôsob akým medzi sebou jednotlivé moduly systému komunikujú. *InputImage* a *DetectedObject* sú objekty dátových tried uchovávajúce informácie o vstupnom snímku a detegovanom objekte.

## Spracovanie vstupu

Spracovanie vstupu predstavuje bod, v ktorom dáta z kamier vstupujú do systému. Spoločne s načítaním snímok zabezpečuje tiež odovzdávanie základných informácií o snímkoch ako sú názvy zdrojov snímok a ich rozlíšenie. Umožňuje tiež úpravu tohto rozlíšenia (zmenu veľkosti snímku). Poskytuje rovnako možnosť načítavať všetky snímky zo vstupu alebo len každý n-tý snímok. Táto možnosť spoločne so zmenou veľkosti snímku dokáže v značnej miere zrýchliť výpočty v prípadoch, kedy nám je poskytnutých viac dát ako k detekcii potrebujeme. Napríklad, ak je zdroj snímok natočený snímkovacou rýchlosťou 30 snímok za sekundu, tak je zbytočné vykonávať detekciu na všetkých týchto snímkoch, keďže sa medzi sebou nebudú veľmi líšiť. Pri zvolení načítavania každého tretieho snímku trojnásobne znižujeme náročnosť výpočtov, pričom si stále zachovávame kvalitnú snímkovacu rýchlosť 10 snímok za sekundu. V systéme sú implementované dve triedy spracovania vstupu a to spracovanie vstupu z jedného súboru a spracovanie dát z datasetu WILDTRACK [5]. Spracovanie dát z datasetu WILDTRACK synchronizovane načítava snímky až zo siedmich kamier. Obe implementácie demonštrujú flexibilitu implementovaného systému. Výstup tohto modulu predstavujú dátové objekty *InputImage*, do ktorých sú uložené načítané snímky zo vstupných zdrojov, názov kamery, z ktorej boli načítané, spolu s časovou známou vyhotovenia snímku.

## Detekcia osôb

Detekcia osôb je v súčasnej implementácii reprezentovaná konvolučnou neurónovou sieťou Mask R-CNN (kapitola 2.2.4). Na vstupných snímkoch identifikuje objekty. Sú vybrané len objekty, ktoré sú klasifikované ako ľudia s pravdepodobnosťou väčšou ako predom stanovený prah. Táto neurónová sieť spoločne s klasifikáciou objektu poskytuje aj masku pod ktorou sa nachádza detegovaný objekt. Za využitia tejto masky je následne z detekcia človeka vypočítaný histogram farby jeho oblečenia. Tento histogram je spolu s pozíciou ohraničujúceho obdĺžnika, názvu kamery a časovej známky uložený do objektu dátovej triedy *DetectedObject*.

## Zistenie pozície osôb

Zistenie pozície osôb je implementáciou posledného kroku algoritmu PnP (kapitola 2.3). Pri inicializácii je na základe cesty špecifikovanej v konfiguračnom súbore načítaný slovník, v ktorom kľúč do slovníka predstavuje názov kamery a hodnota v slovníku konfiguračnú maticu. Predchádzajúcim krokom sú poskytnuté objekty dátovej triedy *DetectedObject*, ktoré reprezentujú jednotlivé detekcie. Pre každú detekciu je vybraný bod, ktorý predstavujúci kontakt danej osoby so zem. Tento bod je stred spodnej hrany ohraničujúceho obdĺžnika detekcie. Je znázornený ako červený bod na obrázku 4.4. Po získaní súradníc  $x$  a  $y$  tohto bodu v súradnicovom priestore kamery je implicitne zvolená výška bodu  $z = 1$ . Takto vytvorená matica v tvare

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

je vynásobená konfiguračnou maticou. Z načítaného slovníku je vybraná správna konfiguračná matica podľa názvu kamery z príslušného objektu *DetectedObject*, pre ktorý je počítaná poloha. Získané súradnice sú spätne uložené do daného objektu.



Obr. 4.4: Výstup detekcií neurónovej siete Mask R-CNN. Farba ohraničujúceho obdĺžnika zodpovedá dominantnej farbe histogramu oblečenia danej detekcie. Červená bodka v strede dolnej hrany ohraničujúceho obdĺžnika predstavuje bod, podľa ktorého je počítaná pozícia objektu v monitorovanom priestore.

### 4.3 Identifikácia a sledovanie osôb

Vstup tohoto modulu predstavujú jednotlivé detekcie s priradenou pozíciou v monitorovanom objekte. Pri inicializácii modulu sú načítané rôzne prahové hodnoty z konfiguračného súboru. Tieto hodnoty boli testovaním optimalizované. Pre rôzne vstupné zdroje snímkov boli nájdené rôzne optimálne hodnoty. Vyňatím týchto hodnôt do konfiguračného súboru je zabezpečená ich ľahká meniteľnosť pri aplikovaní systému do rôznych prostredí.

V tomto module sú udržiavané informácie o všetkých detegovaných osobách. Konkrétne sa jedná o ich poslednú pozíciu, čas kedy boli naposledy detegované, z ktorých kamier boli naposledy videné, ich trajektóriu, histogram farby a všetky predchádzajúce pozície.

Ako prvý krok pri identifikácii osôb bol implementovaný algoritmus zo štúdie [33]. Princíp tohoto algoritmu spočíva v zlučovaní detekcií z viacerých kamier tak, ako to je popísané v kapitole 2.3. Po spustení tohoto algoritmu sú získané skupiny detekcií so spoločnou polohou, pri ktorých je vysoká pravdepodobnosť, že patria tej istej osobe.

Následne je zostavená matica vzdialeností každej skupiny detekcií ku každej z existujúcich osôb. V module sa nachádza konštanta *fill\_distance*, ktorá je vložená do matice na príslušné miesto, ak daná kombinácia detekcie a osoby nespĺňa jednu z požiadaviek. Požiadavky sú nasledovné:

- **Časová lokalita:** ak rozdiel medzi aktuálnou časovou značkou a poslednou časovou značkou, na ktorej bola osoba videná je väčší ako *časový prah*, osoba nie je zvažovaná ako potenciálny kandidát a do matice je vpísaná hodnota *fill\_distance*. Táto kontrola by mala zabrániť priradeniu detekcií osobám, ktoré už opustili monitorovanú oblasť.

- **Priestorová lokalita:** pridelenie osoby detekcii je zvažované len vtedy ak euklidovská vzdialenosť medzi pozíciou skupiny detekcií a pozíciou osoby je menšia ako *vzdialenostný prah*. V prípade, že máme dostatok predchádzajúcich pozícií osoby na výpočet jej trajektórie, je táto vzdialenosť k aktuálnej pozícii váhovaná s vzdialenosťou k očakávanej pozícii. Táto podmienka odfiltruje všetky osoby, ktoré sú vo väčšej vzdialenosti od skupiny detekcií, než vzdialenosť, ktorá je zvládnuteľná ľudskou bytosťou.
- **Farebná lokalita:** osoba, ktorej Hellingerova vzdialenosť medzi priemerným histogramom farby oblečenia skupiny detekcií a histogramom farby oblečenia osoby je väčšia ako *farebný prah*, nie je zvažovaná ako potenciálny kandidát. Keďže systém počíta s nemennosťou črt jednotlivých osôb, ako je opísaná v kapitole 2.4, touto kontrolou je zabránené priradeniu osoby skupine detekcií s rozdielnou farbou oblečenia.

Hodnota, ktorá sa pri nesplnení ktorejkoľvek z týchto podmienok vloží do matice (*fill\_distance*), je určená podľa použitého algoritmu pridelenia. V prípade použitia Munkresovho algoritmu je táto hodnota nastavená na *sys.maxsize*<sup>5</sup>, ktorá predstavuje najväčšiu možnú hodnotu reprezentovateľnú v jazyku Python. V prípade použitia algoritmu hľadania maximálneho riešenia v párných grafoch je táto hodnota nastavená na 0 (nulu), čo znamená že medzi uzlami neexistuje spojenie.

Takto vygenerovaná matica je spracovaná zvoleným algoritmom pridelenia. Na základe výstupu z tohoto algoritmu sú vytvorené nové osoby (ak nebola skupine detekcií pridelená žiadna osoba) alebo aktualizované údaje už existujúcich osôb (podľa údajov o polohe, farbe oblečenia a pozícii skupiny detekcií).

## Spojenie redundantných osôb

Posledná operácia, ktorá je nad novo vytvorenými/aktualizovanými osobami vykonaná je spojenie redundantných osôb. Počas procesu spájania polôh detekcií alebo priradovania detekcií osobám môžu nastať situácie, v ktorých by skupina detekcií mala byť pridelená už existujúcej osobe ale namiesto toho je vytvorená nová osoba. Tento problém je riešený spájaním novovytvorených osôb do už existujúcich osôb pri splnení niekoľkých podmienok.

Novovytvorenou osobou sa rozumie taká, ktorej história predchádzajúcich polôh má dĺžku 1 a zároveň časová známka tejto polohy je zhodná s aktuálnou časovou známkou snímku. Inak povedané osoba, ktorá bola vytvorená pri spracovávaní aktuálneho snímku.

Za existujúcu osobu vhodnú na spájanie je považovaná taká, ktorej dĺžka histórie predchádzajúcich polôh je väčšia ako 1 a posledná zaznamenaná poloha bola v aktuálnom alebo predchádzajúcom snímku. Tým sa rozumejú osoby, ktoré boli detegované v aktuálnom snímku alebo ktorých detekcia bola v aktuálnom snímku stratená. V prípade, ak sa jedná o stratenú osobu, na základe trajektórie osoby je vypočítaná očakávaná poloha osoby na tomto snímku.

Následne je pre každú novovytvorenú osobu vytvorené usporiadané pole. Toto pole obsahuje Euklidovských vzdialenosti medzi polohou novovytvorenej osoby po polohy už existujúcich osôb. Spojenie osôb je aplikované, ak je vzdialenosť medzi novovytvorenou osobou a existujúcou osobou menšia ako *prah spojenia trajektórie* (ak sa jedná o stratenú osobu) alebo menšia ako *prah spojenia pozície* (ak ide o osobu detegovanú v aktuálnom snímku). Poslednú kontrolu pred spojením osôb predstavuje porovnanie kamier, z ktorých

<sup>5</sup><https://docs.python.org/3/library/sys.html>



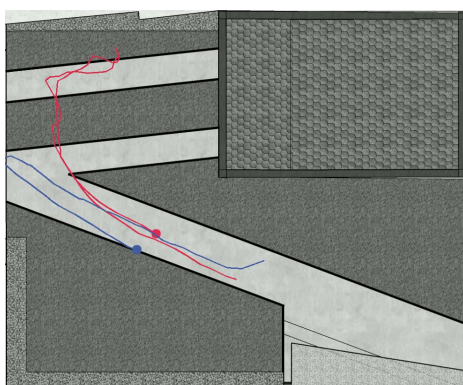
bola existujúca a novovytvorená osoba videná. Keďže jedna osoba nemôže byť videná na jednej kamere dvakrát, spojenie osôb sa uskutoční len, ak neexistuje prienik medzi týmito dvomi množinami kamier. Táto kontrola prieniku kamier neprebehne, pokiaľ existujúca osoba nebola videná na aktuálnom snímku a pracujeme len s jej trajektóriou.

Ak sa spojenie vykoná, všetky parametre osoby sa aktualizujú, pričom nová poloha sa vypočíta ako  $0.7 * \text{pozícia existujúcej osoby} + 0.3 * \text{pozícia novej osoby}$ . Po vykonaní spájania sú vymazané tie novovytvorené osoby, ktoré boli spojené do už existujúcich osôb.

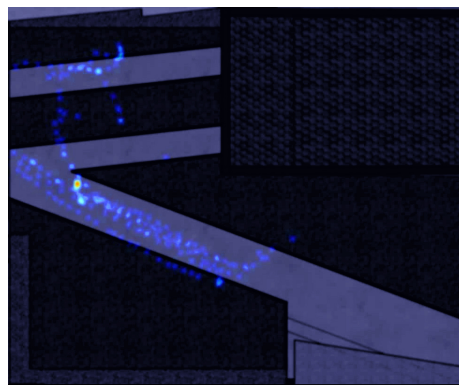
## 4.4 Generovanie výstupu

Výstup aplikácie v podobe tepelnej mapy, trajektórií jednotlivých osôb a štatistík pre jednotlivé expozície je generovaný posledným modulom. Z identifikačného modulu sú poskytnuté všetky detegované osoby, vrátane ich polôh a čít, ktoré sú postupne zanalyzované. Pri spracovávaní týchto pozícií sú generované nasledujúce štatistiky:

- **Mapa trajektórií** zobrazuje trajektórie jednotlivých osôb naprieč monitorovanou oblasťou. Jednotlivé trajektórie predstavujú línie vykreslené na pôdorys monitorovaného objektu. Ak nie je pôdorys k dispozícii, tieto trajektórie sú vykreslené na bielu plochu. Farby jednotlivých línií sú zvolené dynamicky tak, aby reprezentovali dominantnú farbu oblečenia danej osoby. Posledná známa pozícia osoby je vyobrazená väčším kruhom na konci línie spolu s číslom predstavujúcim interný identifikátor osoby. Je vidieť na obrázku [4.5a](#).
- **Tepelná mapa (heatmap)** farebne zvýrazňuje miesta, na ktorých sa návštevníci zdržali najdlhšie. Je vykreslená ako transparentná vrstva na pôdoryse monitorovanej oblasti. V prípade absencie pôdorysu je ako podklad použité čierne pozadie. Najsvetlejšie miesta predstavujú časti, v ktorých návštevníci trávili najviac času. Je vidieť na obrázku [4.5b](#).
- **Štatistika expozície** predstavuje metriky vypočítané pre všetky užívateľom definované expozície. Jedná sa o priťahujúcu silu (pojednávajúcu o pomere návštevníkov, ktorý navštívili exponát voči všetkým návštevníkom múzea) a udržiavajúcu silu (vyjadrujúcu do akej miery si exponát udržal návštevníkovu pozornosť). Tieto metriky sú bližšie popísané v kapitole [2.1](#).



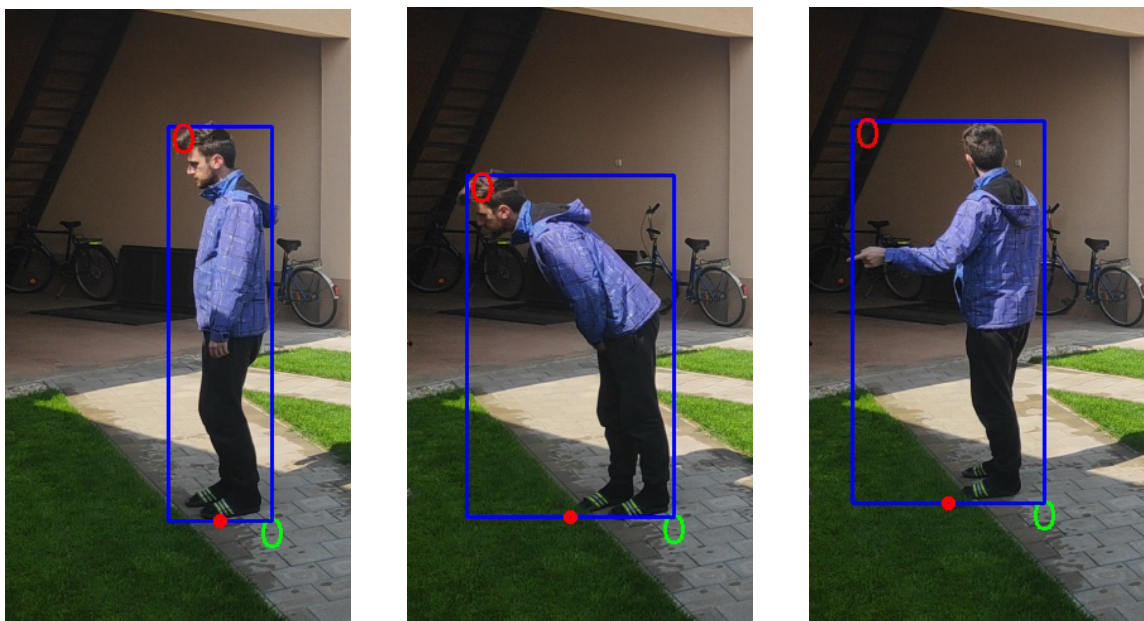
(a) Výstup zobrazujúci trajektórie sledovaných osôb na pôdoryse monitorovanej oblasti.



(b) Výstup zobrazujúci tepelnú mapu miest, na ktorých sledované osoby strávili najviac času. Ako podklad bol využitý pôdorys sledovanej oblasti.

Obr. 4.5: Výstup sledovaných trajektórií 4.5a a tepelnej mapy 4.5b z monitorovanej oblasti, na ktorej prebiehala detekcia.

Pri počítaní štatistiky pre jednotlivé expozície bolo nutné vyriešiť problém oscilácie polohy sledovanej osoby. Môžeme predpokladať, že návštevník nebude pri prezeraní jednotlivých expozícií nehybne stáť ale jeho postoj sa bude meniť. Spoločne so zmenou postoja osoby sa mení aj ohraničujúci obdĺžnik jej detekcie, na ktorom závisí zisťovanie polohy osoby v monitorovanom objekte. Tým pádom sa podľa zmeny postoja bude meniť aj zistená poloha návštevníka, ktorý stojí na mieste, prezerá si expozíciu a hýbe časťami tela. V prípade stanovenia presných hraníc plochy patriacej danému exponátu, z ktorej bude počítaná štatistika, by mohlo dochádzať k opakovanému opúšťaniu a vstupovaniu do tejto plochy osobou stojacou avšak meniacou svoj postoj. Toto vychádzanie a vchádzanie by skresľovalo výsledné štatistiky. Tento problém možno vidieť vizualizovaný na obrázku 4.6. Ak by tráva na tomto obrázku predstavovala detekčnú plochu jedného exponátu a chodník plochu druhého exponátu, vďaka zmene postoja by návštevník prechádzal medzi týmito plochami. Pre vyriešenie tohto problému bol implementovaný princíp hysterézie, teda dvoch plôch, jednej väčšej predstavujúcej plochu pre opustenie a druhej menšej predstavujúcej plochu pre vstúpenie. Vizualizácia tejto hysterézie sa nachádza na obrázku 4.7. Aby osoba bola zaznamenaná do štatistik pre daný exponát musí vstúpiť do jeho vnútornej, menšej, plochy. Avšak na opustenie daného exponátu, osoba musí opustiť väčšiu, vonkajšiu plochu.

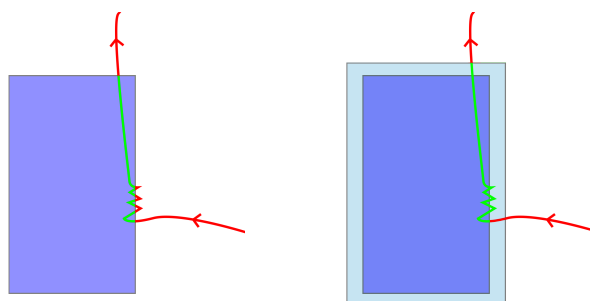


(a) Detekcia osoby stojacej pred exponátom.

(b) Detekcia osoby prezerajúcej si exponát.

(c) Detekcia osoby ukazujúcej na exponát.

Obr. 4.6: Vizualizácia zmien postoja osoby pri prezeraní si exponátu. V vzpriamenej polohe 4.6a, snažiaci si ho pozrieť zblízka 4.6b a ukazujúci na exponát 4.6c. Pri dvoch posledných pózach dochádza k vyčnievaniu častí tela osoby mimo jej stabilnú pózu a tým aj zmenu ohraničujúceho obdĺžnika spolu s bodom, pre ktorý je počítaná poloha osoby (červená bodka na spodnej hrane obdĺžnika).



(a) Bez použitia hysterézie

(b) S využitím hysterézie

Obr. 4.7: Vizualizácia zbierania štatistík pre expozíciu bez využitia hysterézie a s jej využitím. Modré obdĺžniky reprezentujú plochu pred exponátom, z ktorej sú zbierané štatistiky. Červeno-zelená čiara predstavuje zaznamenanú trajektóriu osoby. Podľa toho či osoba bola počítaná do štatistík expozície sa mení farba trajektórie: zelená ak sa počíta do štatistiky, červená ak sa nepočíta. Na obrázku bez použitia hysterézie 4.7a možno vidieť opakované vstupovanie a vychádzanie osoby z definovanej plochy. Toto opakované vychádzanie je eliminované využitím hysterézie 4.7b. Osoba je počítaná do štatistiky až po vstupe do vnútornej zóny, tmavého obdĺžnika. Osoba opúšťa plochu daného exponátu až výstupom z vonkajšej zóny, bledého obdĺžnika.

Modul vyžaduje na generovanie štatistík pre jednotlivé expozície konfiguračný súbor popisujúci každú zo sledovaných expozícií. Sú vyžadované údaje o odhadovanom čase na „vstrebávanie informácií“ ako je popísané v kapitole 2.1 a súradnice vstupnej a výstupnej plochy. Tento súbor je vo formáte json a je načítavaný pri inicializácii štatistík.

## 4.5 Užívateľské rozhranie

Konfiguračná časť systému si vyžaduje užívateľov vstup na zadávanie pozície bodov či výber správneho snímku na vykonanie konfigurácie. Z tohto dôvodu obsahuje jednoduché užívateľské rozhranie. Pri výbere konfiguračnej metódy oamotovaných bodov je nutné zadať presné súradnice bodov v súradnicovom priestore kamery. Na uľahčenie tejto činnosti je implementovaná funkcia spúšťaná pri kliknutí ľavého tlačidla myši, ktorá zaznamená súradnice v snímku pod kurzorom myši a vyzve užívateľa na zadanie súradníc v súradnicovom priestore monitorovaného objektu. Následne súradnice v oboch súradnicových priestoroch uloží do pamäte. Na overenie správnosti konfigurácie je v podobnom zmysle implementovaná funkcia spúšťaná pri kliknutí pravého tlačidla myši. Táto funkcia vypíše súradnice bodu pod kurzorom myši v oboch súradnicových priestoroch. Pre vizuálnejšie overenie správnosti konfigurácie je implementovaná funkcia vykresľujúca vypočítanú rovinu predstavujúcu zem monitorovaného objektu. Je vykresľovaná pomocou červených bodov umiestnených každých 20 cm po oboch osiach roviny. Poskytuje vhodnú vizualizáciu správnosti konfigurácie, keďže je na rovine dobre viditeľné jej naklonenie. Spolu s týmito funkciami na zjednodušenie práce a vizualizáciu riešenia, je užívateľovi poskytnuté rozhranie v podobe ovládania prehrávania snímkov a volania jednotlivých funkcií konfigurácie podľa stlačených tlačidiel klávesnice.

Samotný systém v sebe obsahuje užívateľské rozhranie primárne určené na ovládanie prehrávania obrazových dát, na vyžiadanie výstupu aplikácie a na preskúmavanie existujúcich detekcií. Pri spúšťaní systému je možnosť vybrať si, či má byť systém spustený s užívateľským rozhraním alebo bez neho, pre rýchlejšie spracovávanie dát. Užívateľské rozhranie spočíva v zobrazovaní aktuálne spracovávaných snímkov v novom okne. Pre vizualizáciu práce systému sú pri zobrazení užívateľského rozhrania vykresľované ohraničujúce obdĺžniky detegovaných osôb spolu s číslom identifikujúcim poradie detekcie neurónovou sieťou Mask R-CNN a identifikačným číslom osoby. Po kliknutí ľavým tlačidlom myši na osobu je zobrazené nové okno s podrobnými informáciami o danej osobe a to: jej identifikátor, pozícia v rámci kamery a taktiež súradnicového priestoru monitorovanej oblasti, názvy kamier, na ktorých bola naposledy zachytená, spolu s histogramom farby jej oblečenia a ďalšími informáciami.

# Kapitola 5

## Testovanie

Testovanie implementovaného systému prebiehalo na dvoch setoch dát. Pre vyhodnotenie úspešnosti detegovania a sledovania ľudí bol systém testovaný na datasete WILDTRACK [5]. Keďže má tento dataset v sebe obsiahnuté veľké množstvo dát, pre prehľadnejšie testovanie generovania výstupu aplikácie boli použité vlastné nahrávky vyhotovené v kontrolovanom prostredí.

### 5.1 Využitie datasety

**WILDTRACK** dataset obsahuje zábery z rušného prostredia zaplneného ľuďmi. Konkrétne boli zábery zachytené pred hlavnou budovou univerzity ETH v Zürichu počas slnečného počasia. Dataset obsahuje pohľady zo siedmych kamier, ktorých zorné uhly sa z veľkej časti prekrývajú. Presnejšie boli použité tri GoPro Hero 4 a štyri GoPro Hero 3 kamery. Zo záberov bola odstránená deformácia objektívu a zábery zo všetkých kamier boli zosynchronizované. Spolu je poskytnutých 400 záberov pri snímkovacej frekvencii 2 snímky za sekundu. Snímky sú v rozlíšení FullHD teda 1920x1080 pixlov. V priemere sa na jednom snímku jednej kamery nachádza 20 ľudí čo spolu poskytuje približne 56 000 jednotlivých detekcií ( $400 * 20 * 7$ ). Zosynchronizovaný pohľad na snímky je možno vidieť na obrázku 5.1. Ku každej kamere sú poskytnuté aj externé a interné kalibračné dáta, z ktorých sa dá za pomoci algoritmu PnP určiť presná pozícia kamier. Dataset mimo snímkov taktiež obsahuje ručne oannotované dáta. Tieto dáta pozostávajú z pravdivých detekcií osôb spoločne s ich identifikátormi, pozíciou v rámci súradnicového systému jednotlivých kamier ako aj pozíciou v rámci monitorovanej oblasti.



Obr. 5.1: Zosynchronizovaný pohľad na snímky zo všetkých siedmych kamier datasetu WILDTRACK.

**Vlastné zábery** boli zaznamenané na záhrade rodinného domu. Cieľom týchto záberov je na jednoduchom príklade demonštrovať všetky kľúčové aspekty implementovaného systému. Kalibrácia kamier je vykonaná na základe kľúčových bodov sledovanej oblasti, konkrétne vchodu do budovy a „kalibračnej krabice“ (obrázok 4.1). Na záberoch sa objavujú dve osoby, ktoré sa voľne pohybujú v monitorovanom priestore. K dispozícii je taktiež pôdorys sledovanej oblasti, ktorý bol využitý pri konfigurácii a rovnako aj pri generovaní výstupu.

## 5.2 Výsledky testovania na datasete WILDTRACK

Výkonnosť implementovaného riešenia bola porovnávaná s podobnými riešeniami, ktoré boli rovnako testované na datasete WILDTRACK. Boli hodnotené podľa dvoch vlastností. Prvou z nich je schopnosť detegovať objekt, ktorú hodnotia metriky MOTChallenge [21]. Druhou vlastnosťou je sledovanie zachytených objektov naprieč časom, ktorého presnosť je hodnotená podľa metrík popísaných v štúdiu [18]. Presnejšie sú v tabuľkách 5.2 a 5.3 použité nasledovné metriky:

**FP, FN, TP, GT** (False Positive, False Negative, True Postives, Grand Total) predstavujú detekcie, ktoré boli zachytené avšak v skutočnosti neexistujú (FP), detekcie, ktoré neboli zachytené avšak v skutočnosti existujú, teda boli „stratené“ (FN), správne označené detekcie (TP) a celkový počet detekcií (GT)

**Presnosť, spätné vybavenie** sa vypočíta podľa nasledovných vzorcov:

$$\text{Presnosť} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Spätné vybavenie} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

**F-Skóre** reprezentuje harmonický priemer medzi *presnosťou* (precision) a *spätné vybavenia* si (recall) detekcií.

**MODA** (Multiple Object Detecion Accuracy) označuje presnosť detekcie viacerých objektov a vypočíta sa podľa vzorca:

$$\text{MODA} = 1 - \frac{\sum_t \text{FN}_t + \text{FP}_t}{\sum_t \text{GT}_t}$$

kde  $t$  predstavuje index snímky. Reprezentuje presnosť s akou boli objekty identifikované.

**MODP** (Multiple Object Detecion Precision) vyjadruje presnosť lokalizácie detekcií teda s akou presnosťou bola určená poloha objektov v scéne a dá sa vypočítať ako:

$$\text{Prekryv} = \sum_{t,i} d_{t,i}$$

$$\text{MODP} = \frac{\text{Prekryv}}{\sum_t \text{TP}_t}$$

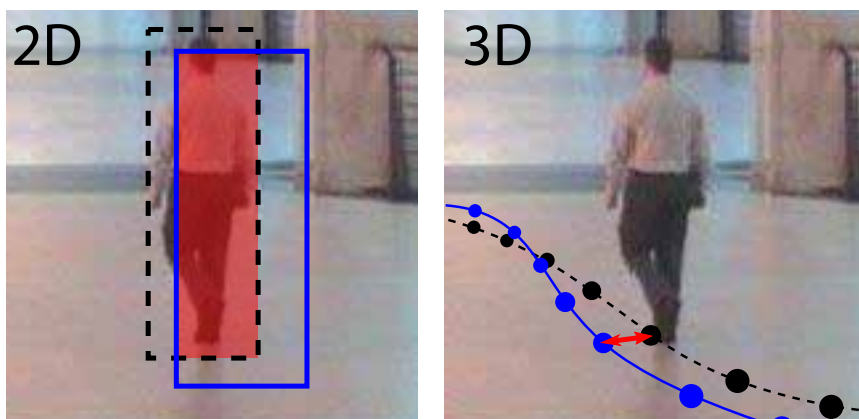
kde  $d_{t,i}$  predstavuje vzdialenosť medzi detekciou  $i$  a jej priradenou anotáciou. Bližšie vizualizovaný prekryv možno vidieť na obrázku 5.2.

**MOTA** (Multiple Object Tracking Accuracy) vyjadruje presnosť sledovania objektov naprieč snímkami. Zohľadňuje výmeny identít medzi jednotlivými detekciami a rovnako tiež priradenie neexistujúcich detekcií (False Positives) a nenájdenie existujúcich detekcií (False Negatives) a počíta sa podľa vzorca:

$$\text{MOTA} = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t}$$

kde  $t$  vyjadruje index snímky a  $IDSW$  (Identity switch) počet razov, kedy detekcia zmenila svoju anotáciu. Vizualizáciu výmeny identity možno vidieť na obrázku 5.3.

**MT, PT a ML** (Mostly tracked, partially tracked, mostly lost) predstavujú počet trajektorií, ktoré boli sledované aspoň 80%, 79-20% a menej ako 20% zo svojej celkovej dĺžky.



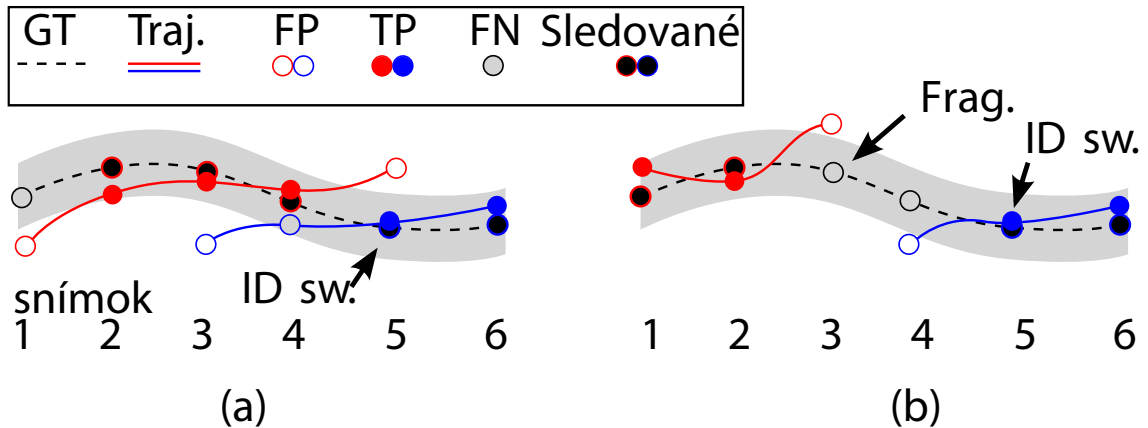
Obr. 5.2: Vizualizácia počítania metriky MODP. Pre metódy, ktoré spracovávajú iba snímky bez anotácie pozícií sa *prekryv* počíta ako pomer plochy prieniku ohraničujúceho obdĺžniku detegovaného porovnanou metódou k ploche oantovaného ohraničujúceho obdĺžniku. Pri metódach, ktoré používajú k identifikácii aj údaje o polohe objektov sa táto hodnota môže počítať ako euklidovská vzdialenosť medzi zistenou polohou a oantovanou polohou<sup>1</sup>.

Implementovaný systém bol porovnaný s niekoľkými poprednými algoritmi. Presnejšie s POM-CNN [25], DeepMCD [6], RCNN-Projected [30] a Deep-Occlusion [2]. Metóda DeepMCD obsahuje viacero variant. Tieto varianty využívajú rôznych modelov a to: GoogLeNet [29], ResNet-18 [12] a DenseNet-121 [14]. jednotlivé varianty s príslušnými tréningovými datasetmi sú popísané v tabuľke 5.1.

Tabuľka 5.1: Varianty DeepMCD s tréningovým datasetom a modelom, ktorý využívajú.

Metóda	Tréningový dataset	Model
Pre-DeepMCD	PETS	GoogLeNet
Top-DeepMCD	WILDTRACK	GoogLeNet
ResNet-DeepMCD	WILDTRACK	ResNet-18
DenseNet-DeepMCD	WILDTRACK	DenseNet-121

<sup>1</sup>Obrázok pochádza zo štúdie [21]



Obr. 5.3: Na obrázku možno vidieť vizualizované jednotlivé prípady sledovania trajektórií, ktoré spracovávajú použité metriky. Na obrázku (a) možno vidieť výmenu identity osoby, kedy sa vymenila červená detekcia za modrú detekciu. Na obrázku (b) možno vidieť fragmentáciu trajektórie. Na snímku (frame) 3 strácame detegovanú osobu, ktorú neskôr znovu nachádzame. Toto má za následok mimo výmeny identity aj stratu presnej trajektórie osoby<sup>2</sup>.

Ako možno vidieť v tabuľke 5.2, navrhovaný systém si v detegovaní osôb vo videozázname počínal veľmi dobre. Prekonali ho len algoritmy, ktoré využívali WILDTRACK dataset nielen na vyhodnocovanie výsledkov ale rovnako aj na tréning. Tento výsledok sa dal čakať, keďže náš systém využíva neuronovú sieť, ktorá bola tréningovaná na COCO datasete [23]. Avšak spomedzi netréningovaných metód, teda metód, ktoré na tréning nepoužívali dataset WILDTRACK, si počínal veľmi dobre.

Tabuľka 5.2: Výsledky testov [5] detekcií rozdielnych metód na datasete WILDTRACK. *Horná časť* oddelená prerušovanou čiarou obsahuje metódy, ktoré boli tréningované na datasete WILDTRACK. *Spodná časť* zobrazuje metódy, ktoré boli tréningované na iných datasetoch ako WILDTRACK.

Metóda	F-Skóre	MODA	MODP
Deep-Occlusion	0.87	0.74	0.54
ResNet-DeepMCD	0.83	0.68	0.64
DenseNet-DeepMCD	0.80	0.64	0.67
Top-DeepMCD	0.79	0.60	0.64
-----	-----	-----	-----
Navrhovaný systém	0.69	0.40	0.63
POM-CNN	0.63	0.23	0.31
RCNN-projected	0.53	0.11	0.18
Pre-DeepMCD	0.52	0.33	0.53

V nasledujúcej tabuľke 5.3 je vidieť porovnanie výsledkov sledovania objektov. Implementovaný systém si v detegovaní objektov počínal horšie ako metóda Deep-Occlusion, s ktorou je porovnávaný. Preto aj v „globálnych“ metrikách, ktoré berú ohľad na všetky

<sup>2</sup>Obrázok pochádza zo štúdie [21]



osoby vyskytujúce sa na snímkach (metrika MOTA), si počína výrazne horšie. Nedokážeme sledovať osobu, ktorú sme nezachytili. Avšak pri porovnaní dĺžky sledovania a jej stability dosahuje podobné výsledky.

Tabuľka 5.3: Výsledky testov [5] sledovania objektov metódy Deep-Occlusion a nášho systému na datasete WILDTRACK.

Metóda	MODA	MOTA	MT	PT	ML
Deep-Occlusion+KSP	0.75	0.69	49 (29%)	79 (46%)	43 (25%)
DeepOcclusion+KSP+ptrack	-	0.72	72 (42%)	74 (43%)	25 (15%)
Navrhovaný systém	0.40	0.25	70 (82%)	14 (17%)	1 (1%)

### 5.3 Výsledky testovania na vlastnom datasete

Testovaním riešenia na datase WILDTRACK bola určená jeho úspešnosť v oblastiach detekcie a sledovania osôb. Testovanie na vlastnom datasete bolo preto primárne zamerané na určenie presnosti konfigurácie a demonštráciu výstupného modulu. Vlastný dataset pozostáva z fotografie a videa, ktoré boli obe zaznamenané na záhrade rodinného domu. Obe boli zaznamenané zo statívu s rovnakými nastaveniami. Video bolo natočené v rozlíšení FullHD (1920x1080) pri snímkovacej frekvencii 25 snímkov za sekundu a má dĺžku 46 sekúnd. Fotografia slúži na kalibráciu kamery a obsahuje umiestnenie „kalibračnej krabice“ v monitorovanom priestore. Video zachytáva pohyb sledovaných osôb. Na zázname sa nachádzajú dve osoby, ktoré sa pohybujú naprieč celým monitorovaným priestorom. Pohybujú sa nepredvídateľne, rozdielnou rýchlosťou, často sa prekrývajú, chodia po vyznačených chodníkoch ale aj mimo nich.

Konfigurácia polohy kamery dosahuje presnosť 10 cm vo vzdialenosti 4 m od počiatku súradnicovej sústavy. Pre generovanie výstupu bol využitý pôdorys monitorovanej oblasti. Tento pôdorys pochádza z 3D modelu sledovanej oblasti, z ktorého boli zisťované aj jednotlivé vzdialenosti konfiguračných bodov. Na obrázku 4.5 je vidieť vykreslený výstup systému na tento pôdorys.

## Kapitola 6

# Záver

Cieľom tejto práce bolo navrhnuť a implementovať systém na sledovanie návštevníkov múzejných expozícií za pomoci kamerovej technológie. Tento cieľ bol splnený a výsledkom je systém, ktorý dokáže spracovať vstupný záznam, detegovať na ňom ľudí, extrahovať ich črty a na základe týchto črt ich identifikovať a sledovať naprieč ďalšími snímkami a sledovaným objektom.

V tejto práci boli zhrnuté výsledky doterajšieho výskumu, na ktorých bol postavený a navrhnutý náš systém. Návrh tohto systému, ktorý z obrazových dát vizualizuje na pôdoryse sledovaného objektu pohyb návštevníkov, bol úspešne implementovaný a testovaním bola táto úspešnosť kvantifikovaná.

Výsledky testovania ukázali, že implementovaný systém dosahuje veľmi dobré výsledky v detegovaní a následnom sledovaní osôb. Presnejšie náš systém dosahuje v priemere 8% zlepšenie v detekcii osôb oproti podobným metódam. Aj napriek tomu sú stále zreteľné nedostatky, prevažne v detekcii a sledovaní čiastočne viditeľných osôb, oproti už existujúcim, lepšie trénovaným metódam.

Implementovaný systém pozostáva z častí, ktoré sú jednoducho vymeniteľné. Vďaka „štandardizovanému“ rozhraniu a formátu prenášaných dát sú jednotlivé časti z veľkej miery samostatné. Táto vymeniteľnosť a samostatnosť jednotlivých modulov zaručuje ľahkú integráciu nových technológií do tohto systému. Pri návrhu systému bolo myslené na jeho budúce využitie v Národnom múzeu v Prahe a preto systém vďaka jeho možnosti jednoduchého prispôsobenia sa bude schopný reagovať na meniace sa podmienky v tomto prostredí.

V budúcnosti pri vyvinutí nových neurónových sietí či lepších algoritmov na sledovanie ľudí bude ich integrácia veľmi jednoduchá, pričom táto výmena zvyšok systému vôbec nezaťažuje. Systém je aplikovateľný nielen v múzeách a galériách ale všade tam, kde sa z ekonomického či bezpečnostného hľadiska oplatí sledovať správanie ľudí. Rovnako existuje potenciál výmeny detekčného modulu a nasadenia systému na sledovanie iných objektov ako ľudí.

# Literatúra

- [1] ALBAWI, S., MOHAMMED, T. A. a AL ZAWI, S. Understanding of a convolutional neural network. In: Ieee. *2017 International Conference on Engineering and Technology (ICET)*. 2017, s. 1–6.
- [2] BAQUÉ, P., FLEURET, F. a FUA, P. Deep occlusion reasoning for multi-camera multi-target detection. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, s. 271–279.
- [3] BEDAGKAR GALA, A. a SHAH, S. K. A survey of approaches and trends in person re-identification. *Image and vision computing*. Elsevier. 2014, zv. 32, č. 4, s. 270–286.
- [4] BOLLO, A. a DAL POZZOLO, L. Analysis of visitor behaviour inside the museum: an empirical study. In: Citeseer. *Proceedings of the 8th international conference on arts and cultural management*. 2005, sv. 2.
- [5] CHAVDAROVA, T., BAQUÉ, P., BOUQUET, S., MAKSAI, A., JOSE, C. et al. WILDTRACK: A Multi-camera HD Dataset for Dense Unscripted Pedestrian Detection. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, s. 5030–5039. DOI: 10.1109/CVPR.2018.00528.
- [6] CHAVDAROVA, T. a FLEURET, F. Deep multi-camera people detection. In: IEEE. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2017, s. 848–853.
- [7] DEMEL, J. *Grafy*. 2. vyd. Libčice nad Vltavou: J. Demel, 2015. ISBN 978-80-260-7684-1.
- [8] DRAHANSKÝ, M., MACEK, I. a GOLDMANN, T. Monitoring of visitors in museum exhibitions. *Journal of the National Museum (Prague), Natural History Series*. 2020, zv. 189.
- [9] DU, J. Understanding of object detection based on CNN family and YOLO. In: IOP Publishing. *Journal of Physics: Conference Series*. 2018, sv. 1004, č. 1, s. 012029.
- [10] GAO, X.-S., HOU, X.-R., TANG, J. a CHENG, H.-F. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*. IEEE. 2003, zv. 25, č. 8, s. 930–943.
- [11] HE, K., GKIOXARI, G., DOLLÁR, P. a GIRSHICK, R. Mask r-cnn. In: *Proceedings of the IEEE international conference on computer vision*. 2017, s. 2961–2969.

- [12] HE, K., ZHANG, X., REN, S. a SUN, J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, s. 770–778.
- [13] HOPCROFT, J. E. a ULLMAN, J. D. Set merging algorithms. *SIAM Journal on Computing*. SIAM. 1973, zv. 2, č. 4, s. 294–303.
- [14] HUANG, G., LIU, Z., VAN DER MAATEN, L. a WEINBERGER, K. Q. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, s. 4700–4708.
- [15] HUANG, J., RATHOD, V., SUN, C., ZHU, M., KORATTIKARA, A. et al. Speed/accuracy trade-offs for modern convolutional object detectors. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, s. 7310–7311.
- [16] IOANNOU, Y. A. *Structural priors in deep neural networks*. 2018. Dizertačná práca. University of Cambridge.
- [17] KARP, R. M. Kombinatorika, zložitost' a náhodnost. *Pokroky matematiky, fyziky a astronomie*. Jednota českých matematiků a fyziků. 1989, zv. 34, č. 6, s. 313–335.
- [18] KASTURI, R., GOLDFOF, D., SOUNDARARAJAN, P., MANOHAR, V., GAROFOLO, J. et al. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. IEEE. 2008, zv. 31, č. 2, s. 319–336.
- [19] KAUR, H. a SOHI, N. A study for applications of histogram in image enhancement. *Int. J. Eng. Sci.* 2017, zv. 6, č. 6, s. 59–63.
- [20] LANIR, J., KUFLIK, T., SHEIDIN, J., YAVIN, N., LEIDERMAN, K. et al. Visualizing museum visitors' behavior: Where do they go and what do they do there? *Personal and Ubiquitous Computing*. Springer. 2017, zv. 21, č. 2, s. 313–326.
- [21] LEAL TAIXÉ, L., MILAN, A., REID, I., ROTH, S. a SCHINDLER, K. Motchallenge 2015: Towards a benchmark for multi-target tracking. *ArXiv preprint arXiv:1504.01942*. 2015.
- [22] LEPETIT, V., MORENO NOGUER, F. a FUA, P. Epn: An accurate o (n) solution to the pnp problem. *International journal of computer vision*. Springer. 2009, zv. 81, č. 2, s. 155–166.
- [23] LIN, T.-Y., MAIRE, M., BELONGIE, S., HAYS, J., PERONA, P. et al. Microsoft coco: Common objects in context. In: Springer. *European conference on computer vision*. 2014, s. 740–755.
- [24] LIU, C., GONG, S., LOY, C. C. a LIN, X. Person re-identification: What features are important? In: Springer. *European Conference on Computer Vision*. 2012, s. 391–401.
- [25] LONG, J., SHELHAMER, E. a DARRELL, T. Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, s. 3431–3440.

- [26] MUNKRES, J. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*. SIAM. 1957, zv. 5, č. 1, s. 32–38.
- [27] O'SHEA, K. a NASH, R. An introduction to convolutional neural networks. *ArXiv preprint arXiv:1511.08458*. 2015.
- [28] RABBANI, Q., KHAN, A. a QUDDOOS, A. Modified Hungarian method for unbalanced assignment problem with multiple jobs. *Applied Mathematics and Computation*. Elsevier. 2019, zv. 361, s. 493–498.
- [29] SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S. et al. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, s. 1–9.
- [30] XU, Y., LIU, X., LIU, Y. a ZHU, S.-C. Multi-view people tracking via hierarchical trajectory composition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, s. 4256–4265.
- [31] YU, C., LI, X., DOU, L., LI, J., ZHANG, Y. et al. Implement and Optimization of Indoor Positioning System Based on Wi-Fi Signal. In: Springer. *International Conference on Algorithms and Architectures for Parallel Processing*. 2016, s. 220–228.
- [32] ZHA, H., HE, X., DING, C., SIMON, H. a GU, M. Bipartite graph partitioning and data clustering. In: *Proceedings of the tenth international conference on Information and knowledge management*. 2001, s. 25–32.
- [33] ZHU, C. *Multi-camera people detection and tracking*. 2019.