



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**VYUŽITÍ NEURONOVÝCH SÍTÍ PRO FÚZI  
OBRAZOVÝCH A NEOBRAZOVÝCH DAT**

EXPLOITATION OF NEURAL NETWORKS FOR FUSION OF IMAGE AND NON-IMAGE DATA

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. BOŘEK REICH**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**prof. Dr. Ing. PAVEL ZEMČÍK**

BRNO 2022

## Zadání diplomové práce



Student: **Reich Bořek, Bc.**  
Program: Informační technologie a umělá inteligence  
Specializace: Inteligentní systémy  
Název: **Využití neuronových sítí pro fúzi obrazových a neobrazových dat**  
**Exploitation of Neural Networks for Fusion of Image and Non-Image Data**  
Kategorie: Umělá inteligence  
Zadání:

1. Nastudujte dostupnou literaturu na téma neuronové sítě a jejich využití ve zpracování obrazu a jiných 2D, případně 3D dat. Nastudujte též dostupný software pro tyto účely.
2. Vyberte (po konzultaci se vedoucím práce) vhodnou aplikaci pro fúzi informace, například obraz scény a její "point cloud" obsah sejmутý milimetrovým radarem.
3. Vyberte vhodnou metodu pro implementaci a nastudujte její vlastnosti, popište možnosti řešení.
4. Implementujte vybranou metodu a demonstřujte její vlastnosti na vhodném příkladu.
5. Diskutujte dosažené vlastnosti a možnosti pokračování v práci.

Literatura:

- Dle pokynů vedoucího práce.

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3 zadání

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

Vedoucí práce: **Zemčík Pavel, prof. Dr. Ing.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2021

Datum odevzdání: 29. července 2022

Datum schválení: 3. listopadu 2021

## Abstrakt

Tato diplomová práce se zabývá využitím konvolučních neuronových sítí pro fúzi obrazových a neobrazových dat. V práci jsou zkoumány jak detekční systémy využívající principy hlubokého učení spoléhající se pouze na obrazová data (snímky z kamery), tak detekční systémy, které využívají obrazová i neobrazová data (snímky z kamery a data z milimetrového radaru). Pro účely porovnání těchto typů metod (metod využívajících fúzi a metod spoléhajících se pouze na obrazová data) byla vytvořena jedinečná datová sada obsahující nezpracovaná data z milimetrového radaru a odpovídající časově synchronizované snímky z kamery zaměřená na monitorování dopravy. Dále je v této práci navržena časově synchronizační metoda pro milimetrový radar a kameru s použitím volně dostupného hardware. Za použití vytvořené datové sady je pak ověřena detekční schopnost systému využívajícího pouze data z kamery a fúzního systému využívajícího data z milimetrového radaru i kamery.

## Abstract

This master thesis uses convolutional neural networks to fuse image and non-image data. Both deep learning detection systems that rely only on image data (images from the camera) and that use both image and non-image data (images from the camera and data from the millimeter-wave radar) are studied in this thesis. A unique dataset containing raw millimeter-wave radar data and corresponding time-synchronized images from the camera was created for the purpose of comparing these two types of methods (data fusion methods and methods that utilize only image data). Furthermore, a time synchronization method for millimeter-wave radar and cameras using only off-the-shelf hardware is proposed. Finally, the created dataset is used to verify the detection capability of the system that uses only camera data and the fusion system that uses both millimeter-wave radar and camera data.

## Klíčová slova

počítačové vidění, hluboké učení, strojové učení, fúze dat, fúze informací, detekce objektů, konvoluční neuronová síť, kamera, milimetrový radar, obrazová data, neobrazová data

## Keywords

computer vision, deep learning, machine learning, data fusion, object detection, convolutional neural network, camera, millimeter-wave radar, image data, non-image data

## Citace

REICH, Božek. *Využití neuronových sítí pro fúzi obrazových a neobrazových dat*. Brno, 2022. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Dr. Ing. Pavel Zemčík

# Využití neuronových sítí pro fúzi obrazových a neobrazových dat

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana prof. Dr. Ing. Pavla Zemčíka. Další informace mi poskytl pan Ing. Michal Hradiš Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Bořek Reich  
2. srpna 2022

## Poděkování

Tímto bych chtěl poděkovat mé přítelkyni Bc. Michaelle Matuškové a celé rodině za podporu a trpělivost při psaní práce. Jmenovitě dále děkuji příteli Šimonu Zouharovi, který mi pomohl při získávání datové sady.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Současný stav metod strojového učení</b>	<b>3</b>
2.1	Metody strojového učení . . . . .	3
2.2	Přehled metod hlubokého učení . . . . .	5
2.3	Konvoluční neuronové sítě (CNN) . . . . .	6
<b>3</b>	<b>Současný stav fúzních algoritmů</b>	<b>14</b>
3.1	Dělení fúzních algoritmů . . . . .	14
3.2	Způsoby reprezentace signálů milimetrového radaru . . . . .	18
3.3	Datové sady pro fúzi informací z milimetrového radaru a kamery . . . . .	20
3.4	Postupy časové synchronizace sensorických dat . . . . .	22
3.5	Způsoby kalibrace milimetrového radaru a kamery . . . . .	25
<b>4</b>	<b>Současné postupy zpracování a snímání FMCW radarem</b>	<b>28</b>
4.1	Stanovení vzdálenosti FMCW radarem . . . . .	29
4.2	Stanovení rychlosti FMCW radarem . . . . .	31
4.3	Stanovení směru FMCW radarem . . . . .	33
<b>5</b>	<b>Zhodnocení aktuálního stavu a návrh řešení</b>	<b>38</b>
5.1	Shrnutí dosavadního stavu metod a datových sad pro fúzi sensorů . . . . .	38
5.2	Návrh metody rozpoznávání využívající fúze sensorů . . . . .	39
5.3	Vhodné případy pro ověření navržené metody . . . . .	39
5.4	Technické zadání . . . . .	39
<b>6</b>	<b>Implementace a testování</b>	<b>40</b>
6.1	Popis synchronizace dat z radaru a kamery . . . . .	41
6.2	Způsob získání referenčních výstupních hodnot . . . . .	45
6.3	Popis kalibrace radarových a obrazových dat . . . . .	49
6.4	Popis datové sady a předzpracování . . . . .	54
6.5	Architektura navrženého systému . . . . .	57
6.6	Testování navrženého systému . . . . .	58
<b>7</b>	<b>Závěr</b>	<b>64</b>
	<b>Literatura</b>	<b>65</b>

# Kapitola 1

## Úvod

Počítače potřebují získávat informace ze svého okolí. K tomuto účelu se používají nejrůznější senzory, které mohou získávat data o teplotě, vzdálenosti, obrazová data, či virtuální reprezentaci okolí. Pro počítačové vidění se často využívají kamerové systémy. Kamera však selhává v situacích, kdy je negativně ovlivňována prostředím. Může se jednat o okolnosti jako mlha, smog, špatné světelné podmínky. Jelikož je však možné využít společně s kamerovým systémem i další senzory, lze tato úskalí potlačit a vylepšit schopnosti výsledného systému. Tato práce se zabývá využitím kombinací senzorů pro vylepšení detekce a rozpoznávání.

Hlavní motivací pro výběr tématu zabývajícího se fúzí obrazových a neobrazových dat pro mne byla aktuálnost daného tématu. Počítačové vidění je hojně využíváno v nespočtu odvětví a na vytvářené systémy jsou kladeny velké nároky. Zpracovávané téma jsem také zvolil s ohledem na předchozí zkušenosti s fúzí dat z milimetrového radaru a kamery.

Metody strojového učení jsou v dnešní době všudypřítomné. Mnoho z těchto systémů při tom využívá nejrůznější sensorická data a benefituje z fúze dat. Pro počítačové vidění jsou využívány především kamery, lidary, milimetrové radary, případně hloubkové kamery. Za pomoci dat z těchto senzorů je možné vytvořit virtuální reprezentaci prostředí a zároveň udržovat informaci o pozici agenta. Dále pak umožňují spolehlivě detekovat osoby i v náročných situacích nebo segmentovat obrazová i neobrazová data na objekty a kategorizovat je. Zmíněné metody jsou zajímavými příklady algoritmů, které je možné využít pro nejrůznější případy užití. Lze je použít k navádění autonomních vozidel, monitorování osob, hledání vadných výrobků a mnoha dalším případům užití.

Cílem této diplomové práce je ověřit výhody použití fúzních metod oproti metodám spoléhajícím se pouze na data z jednoho senzoru. Především se pak tato práce zaměřuje na porovnání spolehlivosti fúzních metod oproti metodám pracujícím pouze s daty z kamery ve zhoršených podmínkách. Pro ověření výhod použití fúze dat z více senzorů je v této práci navržen fúzní systém, který je dále porovnán se systémy využívajícími pouze kameru.

Následující kapitoly pojednávají o současném stavu technologií, dostupných prostředcích a metodách souvisejících s problematikou fúze obrazových a neobrazových dat. Jedná se o kapitoly Současný stav metod strojového učení [2](#), Současný stav fúzních algoritmů [3](#) a Současné postupy zpracování a snímání FMCW radarem [4](#). V kapitole Zhodnocení aktuálního stavu a návrh řešení [5](#) je pak zhodnocen prozkoumaný stav a navržen systém vhodný pro demonstraci fúze dat. Další kapitola v pořadí je Implementace a testování [6](#) a pojednává o způsobu řešení stanoveného zadání a zabývá se také testováním vhodných systémů.

## Kapitola 2

# Současný stav metod strojového učení

Shrnutí dosavadního stavu metod strojového učení slouží pro sumarizaci dostupných metod v rámci strojového učení, které jsou pro tuto práci relevantní. Nejedná se tedy o encyklopedický přehled všech metod strojového učení kvůli maximálnímu celkovému rozsahu této práce. Čtenář je zde seznámen s pojmy umělá inteligence, strojové učení, hluboké učení a dále obeznámen s několika typy metod hlubokého učení a jejich příklady, které souvisí s problematikou této práce.

### 2.1 Metody strojového učení

Tato část si klade za cíl seznámit čtenáře s pojmem „strojové učení“, kdy pro uvedení tohoto pojmu do kontextu dále stručně definuje pojmy „umělá inteligence“ a „hluboké učení“. Dále pak rozebírá tradiční techniky strojového učení se zaměřením na specifické vlastnosti těchto metod oproti metodám hlubokého učení.

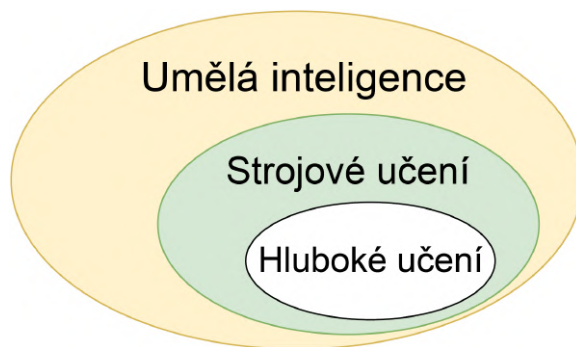
Umělá inteligence je široký pojem, který je definován mnoha způsoby. Zejména kvůli pojmu inteligence, který každý chápe odlišně, je téměř nemožné definovat tento pojem jednotně. Již v roce 1988 byla umělá inteligence popsána jako určité chování stroje, které je totožné s tím lidským [46]. Zejména díky vědeckému pokroku v oblasti informačních technologií se umělá inteligence stala součástí každodenního života mnoha lidí na celém světě. Evropská komise například definuje umělou inteligenci následovně:

*„Umělá inteligence odkazuje na systémy, které projevují inteligentí chování pomocí analýzy svého okolí a vykonáváním akcí, a to s určitým stupněm autonomie, pro dosažení svých cílů.“*

(Evropská komise, 2018, s. 1, vlastní překlad [4])

Pojem „umělá inteligence“ je tedy velmi obecný a metod, které může tento pojem zahrnovat, je mnoho. Kromě jiného do těchto metod spadají metody strojového učení. Vztah zmíněných pojmů je ilustrován na obrázku 2.1.

Co se týče pojmu strojové učení, lze jej opět definovat mnoha způsoby. Andriy Burkov jej ve své knize The Hundred-Page Machine Learning Book definuje následovně:



Obrázek 2.1: Vztah mezi pojmy umělá inteligence, strojové učení a hluboké učení.

*„Strojové učení je obor počítačové vědy zabývající se vytvářením algoritmů, které vyžadují kolekci vzorků určitého fenoménu, aby byly užitečné. Tyto vzorky mohou pocházet z přírody, být generovány jiným algoritmem, nebo jsou ručně vytvořeny člověkem.“*

(Burkov, 2019, s. 3, vlastní překlad [8])

Sám Burkov však ihned pokračuje alternativní definicí, která strojové učení rozděluje do dvou kroků. Prvním je získání datové sady a druhým je algoritmické vytváření statistického modelu na základě dané datové sady [8]. Mezi první, kteří pojem „strojové učení“ použili byl A. L. Samuel, který ve své studii: *Some Studies in Machine Learning Using the Game of Checkers* pojmuje především se schopností učení [45]. Metody strojového učení by tedy měly podle Samuela napodobovat schopnost učení lidí nebo zvířat [45].

Strojové učení lze dělit podle typu datové sady na metody: supervised learning (učení s učitelem), unsupervised learning (učení bez učitele), semi-supervised learning (kombinace učení s učitelem a bez učitele) a reinforcement learning (zpětnovazebné učení) [8].

Metody strojového učení lze tedy považovat za metody využívající poskytnutá data k analýze, učení a následnému rozhodování. Analýza dat typicky spočívá v extrakci a výběru příznaků, které jsou dále použity pro učení a rozhodování. Při extrakci a výběru příznaků jsou vytvořeny, či vybrány příznaky, které maximalizují schopnost učícího algoritmu [22]. Příkladem tradičních metod strojového učení mohou být Support vector machines (SVM), Bayesovské sítě, rozhodovací stromy, či genetické algoritmy [5].

Metody hlubokého učení jsou podmnožinou metod strojového učení. Od tradičních metod strojového učení se odlišují tím, že částečně přeskakují fázi extrakce a výběru příznaků, která může být časově náročná, použitím hlubokých neuronových sítí, které se učí důležité příznaky samy. Není tedy třeba klást takový důraz na výběr a extrakci příznaků, které ku příkladu datovou sadu nejlépe rozdělí na požadované třídy, jako v případě tradičních metod strojového učení.

Dalším rozdílem mezi tradičními metodami strojového učení a metodami hlubokého učení je velikost datové sady. Pro to, aby metody hlubokého učení vykazovaly dobré výsledky je třeba, aby byly učeny na velkém množství dat. Tradiční metody strojového učení si naopak při učení vystačí s menší datovou sadou. Pokud tedy není k dispozici velká datová sada pro řešení problému, tradiční metody strojového učení budou ve většině případů vykazovat lepší výsledky než metody hlubokého učení.

Metody hlubokého učení dále obsahují mnohem více nastavitelných parametrů a provádějí více komplexních operací (například násobení matic) než tradiční metody strojového

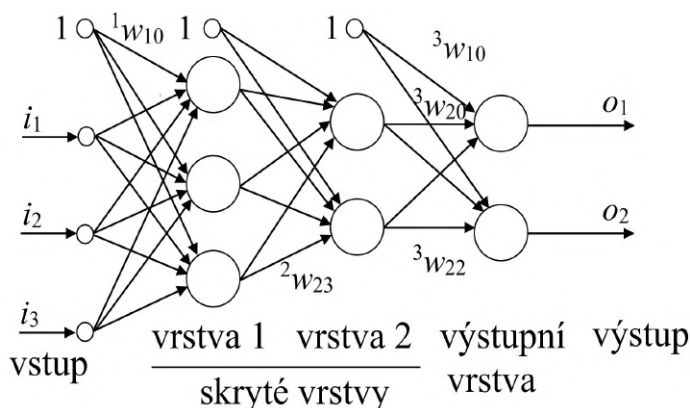


učení. Kvůli tomu je proces učení těchto metod časově i hardwarově náročnější a často vyžaduje vysoce výkonné počítačové komponenty. Nejdůležitější hardwarovou komponentou pro učení hlubokých neuronových sítí je grafická karta, která dokáže mnoho potřebných operací paralelizovat a vykonávat efektivněji než procesor. Pro učení tradičních metod strojového učení ve většině případů postačují méně výkonné hardware prostředky.

## 2.2 Přehled metod hlubokého učení

Tato část se zabývá metodami hlubokého učení se zaměřením na architekturu, postupy učení metod hlubokého učení využívaných při počítačovém vidění a zkoumá především metody využívající data z kamery a milimetrového radaru.

Jak je popsáno výše, metody hlubokého učení se od tradičních metod strojového učení v mnohém liší. Metody hlubokého učení jsou však obecně podmnožinou metod strojového učení, které využívají pro účely extrakce příznaků ze sensorických dat umělé neuronové sítě [5]. Pomocí těchto příznaků dále realizují danou činnost jako klasifikace<sup>1</sup>, detekce<sup>2</sup>, segmentace<sup>3</sup> a jiné. Za metody hlubokého učení se považují metody, které v rámci své architektury využívají umělé neuronové sítě zahrnující skryté vrstvy. Příkladem architektury umělé neuronové sítě je obrázek 2.2.



Obrázek 2.2: Příklad architektury hluboké neuronové sítě se skrytými vrstvami.

Tato neuronová síť je příkladem dopředné plně propojené neuronové sítě využívající umělé neurony obecně definované jako:  $y = g(f(\mathbf{x}, \mathbf{w}))$ , kde  $f$  je básová funkce a  $g$  aktivační funkce. Obecně se však taková neuronová síť může skládat z mnoha různých komponent, v případě konvolučních neuronových sítí to jsou konvoluční neuronové vrstvy nebo pooling vrstvy. Dalším příkladem mohou být paměťové komponenty rekurentních sítí pracujících s na sobě závislými vzory, nebo vysokoúrovňové komponenty enkodér a dekodér pro síť typu enkodér/dekodér [5].

Pro potřeby trénování je pro metody hlubokého učení nejčastěji využíván algoritmus zpětné propagace chyby („backpropagation“), který provádí nastavení parametrů dopředných acyklických neuronových sítí s diferencovatelnou aktivační funkcí [18]. Algoritmus mi-

<sup>1</sup>Vstupnímu obrázku je přiřazena jedna, nebo několik rozpoznávaných tříd [7].

<sup>2</sup>Označení rozpoznávaných objektů ohraničujícím rámečkem a zařazení rámečkem označených objektů do rozpoznávaných tříd [7].

<sup>3</sup>Segmentaci lze definovat jako detekci na úrovni pixelů. Místo ohraničujících rámečků, jsou pixelům vstupních obrazových dat přiřazeny rozpoznávané třídy [7].

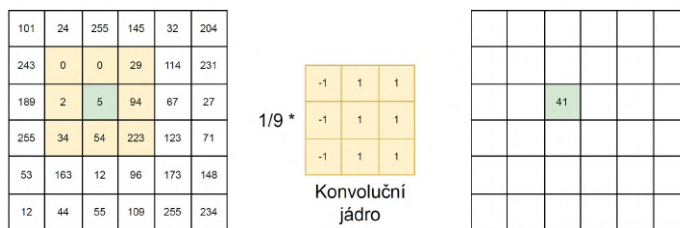
nimalizuje ztrátovou funkci  $L$ , což je funkce všech vah sítě vyjadřující odchylku odezvy sítě od požadovaných hodnot. Příkladem ztrátové funkce jsou křížová entropie pro klasifikaci, či střední kvadratická chyba pro úlohy regrese [18].

## 2.3 Konvoluční neuronové sítě (CNN)

Konvoluční neuronové sítě (CNN) jsou acyklické neuronové sítě skládající se z konvolučních vrstev, aktivačních vrstev, seskupujících (pooling) vrstev a plně propojených vrstev [7]. Primární využití konvoluční neuronové sítě nacházejí při detekci a klasifikaci obrazových dat a ke klasifikaci vět v textech. Hlavními výhodami konvolučních neuronových sítí jsou zachování 2D kontextu dat a sdílení parametrů sítě, což v praxi znamená snížení počtu parametrů neuronové sítě [7].

### Konvoluční vrstva

Konvoluční vrstva získala svůj název podle operace, kterou provádí, tedy konvoluce. Tato operace je dána vztahem:  $I(u, v) = \sum_{i=-a}^b \sum_{j=-c}^d h(i, j)I(u + i, v + j)$ , kde  $(u, v)$  jsou souřadnice pixelu, jehož hodnota má být spočtena a  $a, b, c, d$  definují velikost konvolučního jádra (filtru) [18]. Pro v praxi využívanou velikost konvolučního jádra obvykle platí, že  $a = b = c = d$ . Rozměry jádra je pak možné definovat jako:  $(2a + 1) \times (2a + 1)$ . Operace konvoluce je názorně ukázána na obrázku 2.3.

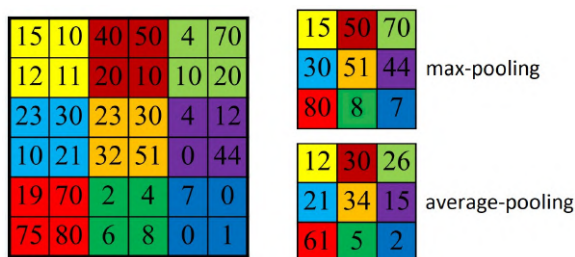


Obrázek 2.3: Ukázka operace konvoluce s konvolučním jádrem (3x3).

Konvoluční vrstva operaci konvoluce provádí s danou hodnotou kroku (S) a výplně (P). Krok S udává posun konvolučního jádra a výplň P počet přidaných sloupců a řádků kolem vstupu. Výplň P tedy zvětšuje výslednou velikost výstupu po aplikaci konvoluce, je tedy díky ní možné zařadit více konvolučních vrstev po sobě bez ztráty informací na okrajích vstupu (obrázku). Krok S pak naopak snižuje velikost výsledku konvoluce, místo redukce dimenzionality pomocí většího kroku se často využívá pooling vrstva [18].

### Pooling vrstvy

Pooling vrstvy slouží ke snížení velikosti vstupu. Existuje několik strategií pro realizaci pooling vrstvy: max-pooling, average-pooling a další [18]. Nejpoužívanější je přístup max-poolingu, kdy se z dané oblasti vybírá nejvyšší hodnota, která se pak stane výstupem. Obvykle používaný krok pooling vrstvy je  $S = 2$  [18]. Ukázka těchto operací je vyobrazena na následujícím obrázku 2.4.



Obrázek 2.4: Ukázka max-pooling a average-pooling operace s krokem  $S = 2$ .

## Aktivační vrstva

V rámci aktivačních vrstev jsou obvykle využívány nelineární funkce, které musí být pro potřeby trénovacího algoritmu backpropagation také diferenciovatelné. Mezi použitelné aktivační funkce patří: schodová funkce, logistická sigmoida, ReLu, hyperbolický tangent ( $\tanh$ ) a další. Aktivační vrstvy slouží k omezení vstupu do daného intervalu, pro zmíněné aktivační funkce jsou to: sigmoida =  $\langle 0; 1 \rangle$ , ReLu =  $\langle 0; \infty \rangle$ , či pro  $\tanh = \langle -1; 1 \rangle$  a další. Za nejčastěji využívanou funkci je považována funkce ReLu daná vztahem:  $f(x) = \max(0; x)$ .

## Plně propojené vrstvy

Plně propojené vrstvy jsou stejné vrstvy jako vrstvy používané ve standardních dopředných sítích využívajících umělé neurony definované v části 2.2 a jejich architektura odpovídá příkladu na obrázku 2.2 [18]. Jejich funkce v rámci CNN může spočívat například v nalezení třídy pro příznaky získané předchozími vrstvami. Aby bylo možné použít výstup předchozí vrstvy, který pracuje s 2D příznaky, jako vstup, je třeba provést operaci „flatten“, která 2D příznaky převede na 1D reprezentaci. Často používanou aktivační funkcí je pro výstupní vrstvu plně propojených vrstev CNN funkce softmax, která konvertuje výstup na hodnoty interpretovatelné jako pravděpodobnosti. Použití funkce softmax je vhodné především pro úlohu rozpoznávání, kdy je pro výstup aktivační funkce možné vybrat třídu s nejvyšší pravděpodobností [18].

## Modely konvolučních neuronových sítí

Za první konvoluční neuronovou síť je považována síť LeNet [26]. Zájem o konvoluční neuronové sítě však vzbudila až síť nazvaná AlexNet [25], která roku 2012 zvítězila v rámci soutěže ImageNet Large Scale Visual Recognition Challenge (ILSVRC) a určila tak trend v metodách počítačového vidění. Od té doby v soutěži ILSVRC zvítězily pouze systémy založené na konvolučních neuronových sítích [44].

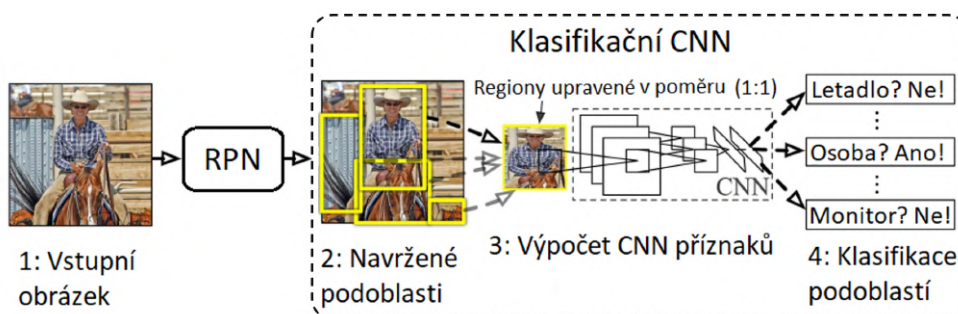
Mezi populární modely CNN pro klasifikaci patří VGG-Net [47], Res-Net [19], GoogLeNet [48], SqueezeNet [21] a další. Mezi CNN modely využívané pro potřeby detekce objektů patří víceprůchodové metody<sup>4</sup> Fast-R-CNN [15], Faster-R-CNN [42] a jednorůchodové metody<sup>5</sup>, jako SSD [30], či různé verze modelu YOLO [37, 38, 39].

<sup>4</sup>Víceprůchodové metody nejprve v obrázku vyhledají oblasti zájmu, které jsou dále klasifikovány pomocí konvoluční neuronové sítě.

<sup>5</sup>Jednorůchodové metody získávají pro vstupní obrázek přímo ohraničení oblastí (bounding boxes) a pravděpodobnosti pro jednotlivé třídy.

## R-CNN, Fast-R-CNN, Faster-R-CNN

R-CNN se po úspěchu sítě AlexNet stala v roce 2014 další úspěšnou metodou využívající konvoluční neuronové sítě. Z R-CNN dále vycházejí oblíbené metody Fast-R-CNN a Faster-R-CNN. R-CNN a její varianty se oproti AlexNet (klasifikace objektů) zaměřují na detekci objektů v obraze [16, 42, 42]. Jedná se o víceprůchodové metody, je tedy možné architekturu této sítě rozdělit na část „region proposal network“ (RPN) a klasifikační CNN. Klasifikační CNN pak provádí extrakci příznaků pouze pro regiony obrázku (podoblasti) navržené pomocí RPN a klasifikaci objektů v nich se nacházejících. Popsané rozvržení je ukázáno na obrázku 2.5.



Obrázek 2.5: Koncepte systému R-CNN [16].

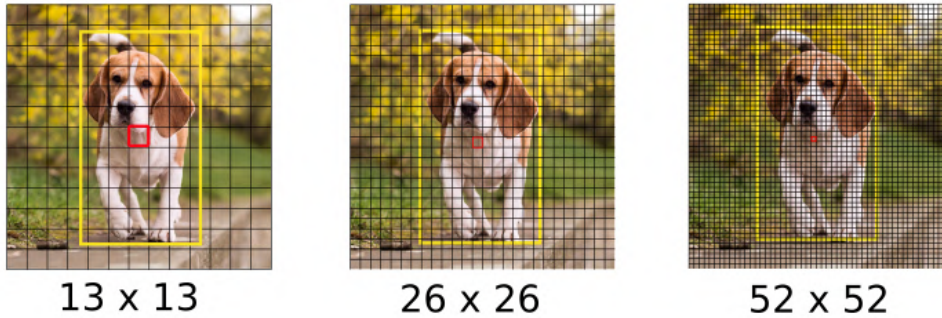
Fast-R-CNN a poté Faster-R-CNN vznikly z toho důvodu, že základní varianta systému (R-CNN) je výpočetně velmi náročná [15, 42]. RPN část R-CNN totiž navrhuje asi 2000 podoblastí, které musí být klasifikační CNN zpracovány. Fast-R-CNN a Faster-R-CNN přináší vylepšení především RPN části architektury sítě. Počet navržených podoblastí je tedy v případě Fast-R-CNN a Faster-R-CNN oproti R-CNN omezen, a tím se snižuje výpočetní náročnost klasifikační části [15, 42]. Snižováním počtu navržených podoblastí obrázku bylo v případě Faster-R-CNN docíleno rychlosti zpracování asi 5 *FPS*, což však stále nedostačuje k detekci objektů v reálném čase [42].

Potenciálním způsobem zrychlení se ukázala změna architektury spočívající ve sloučení částí sítě zodpovědných za návrhy podoblastí (RTN) a klasifikaci podoblastí (klasifikační CNN). Touto úpravou z víceprůchodových metod vznikly metody jednorůchodové, mezi které patří výše zmíněné detekční metody SSD a různé varianty modelu YOLO.

## YOLO

Metody YOLO patří do kategorie jednorůchodových detekčních metod. Jakým způsobem tedy YOLO dokáže určit ohraničení oblasti kolem detekovaného objektu? Modely YOLO se inspirojí modelem Faster-R-CNN, kde je pro získání návrhů oblastí, které jsou dále klasifikovány, použita konvoluční neuronová síť s plně propojenými vrstvami („region proposal network“). Princip modelů YOLO je pak obecně dán jako rozdělení zpracovaného obrázku na buňky, pro které jsou navrženy ohraničující rámečky a pravděpodobnosti rozpoznávaných tříd společně v rámci jedné konvoluční neuronové sítě. Pro první verzi metody YOLOv1 byl obrázek rozdělen na  $(7 \times 7)$  buněk, což však znemožňovalo detekci objektů různých velikostí [37]. Hlavním přínosem metody YOLOv3 je pak řešení tohoto problému předchozích verzí rozdělením obrázku na tři různé velikosti buněk [39]. Rozdělení obrázků podle

YOLOv3 je ukázáno na obrázku 2.6, kde je vstupní obrázek rozdělen na  $(13 \times 13)$ ,  $(26 \times 26)$  a  $(52 \times 52)$  polí [39] oproti rozdělení  $(7 \times 7)$  pro metody YOLOv1 a YOLOv2 [37].



Obrázek 2.6: Rozdělení na buňky pro YOLOv3 [39].

Výstupem modelu YOLO jsou tedy pro každou buňku parametry určující skóre rozpoznávaných tříd a ohraničující rámeček. Pro určení ohraničujícího rámečku by dávalo smysl využít souřadnice středu společně s výškou a šířkou rámečku. Tato reprezentace však přináší problém s nestabilními hodnotami gradientů při trénování. Místo toho jsou použity parametry v logaritmickém prostoru, které jsou společně s přednastavenými ohraničujícími rámečky (anchors), použity pro nalezení reálných ohraničujících rámečků. Výstupem modelu jsou tedy parametry  $t_x, t_y, t_w, t_h$ , které jsou pro nalezení výsledných rámečků využity následovně:

$$b_x = \sigma(t_x) + c_x \quad (2.1)$$

$$b_y = \sigma(t_y) + c_y \quad (2.2)$$

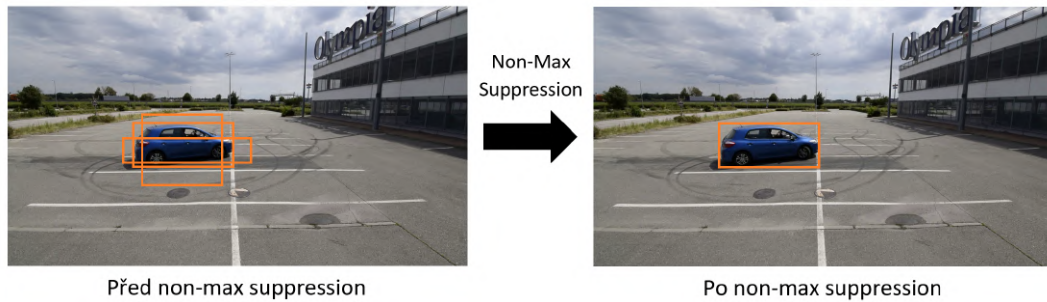
$$b_w = p_w e^{t_w} \quad (2.3)$$

$$b_h = p_h e^{t_h}, \quad (2.4)$$

kde  $b_x, b_y, b_w, b_h$  odpovídají souřadnicím středu, šířce a výšce.  $c_x$  a  $c_y$  jsou pak souřadnice levého horního rohu mřížky a  $p_h, p_w$  jsou rozměry přednastavených ohraničujících rámečků (anchors), které jsou získány analýzou datové sady (ku příkladu datové sady COCO).

Vizualizace zpracovaného výstupu modelu YOLO je ukázána na obrázku 2.7 a označena jako „Před non-max suppression“. Zobrazené ohraničující rámečky jsou výsledkem prahování na základě skóre pro jednotlivé třídy. U zobrazených rámečků je zřejmé, že pouze jeden z nich přesně ohraničuje detekovaný automobil, je tedy třeba z těchto rámečků vybrat ten „nejlepší“. K tomuto účelu slouží algoritmus non-max suppression, který pomocí IoU (Intersection over Union)<sup>6</sup> výběr provede.

<sup>6</sup>Intersection over Union (IoU) odpovídá podílu plochy průniku ohraničujících rámečků a plochy sjednocení rámečků.



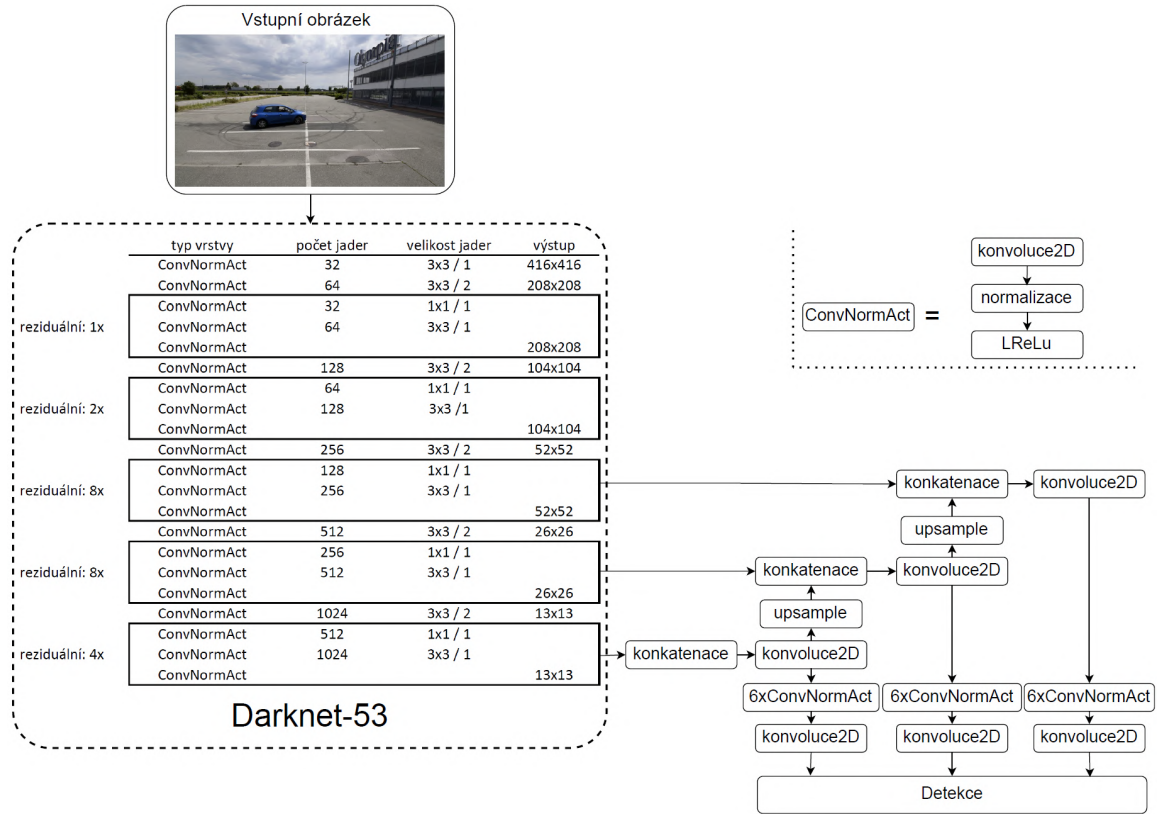
Obrázek 2.7: Ukázka non-max suppression algoritmu.

### Architektura CNN YOLOv3

Na obrázku 2.8 je vyobrazena architektura konvoluční neuronové sítě YOLOv3. Ta se skládá z části pro extrakci příznaků Darknet-53 obsahujících 53 konvolučních vrstev, kdy každou z konvolučních vrstev následuje dávková normalizační vrstva a aktivační vrstva Leaky ReLU<sup>7</sup>, v rámci obrázku 2.8 jsou tyto tři vrstvy označeny jako „ConvNormAct“. Dále pak zahrnuje vrstvy určené pro detekci objektů pro tři různé velikosti. Mezi tyto detekční vrstvy patří konvoluční vrstvy, dávkové normalizační vrstvy, aktivační vrstvy Leaky ReLU (LReLU) a upsample vrstvy.

Vstupem sítě je obrázek o velikosti  $416 \times 416$ . Obrázky s poměrem stran jiným než  $1 : 1$  jsou pro barevné obrázky doplněny o okraje s konstantní hodnotou  $(0; 0; 0)$ , obrázky s velikostí větší než  $416 \times 416$  jsou zmenšeny na odpovídající počet pixelů. Výstupem modelu jsou pak hodnoty definované výše (skóre tříd,  $t_x, t_y, t_w, t_h$ ). Ty jsou pomocí přednastavených ohraničujících rámečků (anchors) a prahování skóre tříd pro jednotlivé ohraničující rámečky interpretovatelné jako detekované objekty.

<sup>7</sup>Leaky ReLU oproti ReLU povoluje malé záporné hodnoty a je definovaná jako:  $f(x < 0) = ax$  a  $f(x > 0) = x$  pro malé hodnoty koeficientu  $a$ , typická hodnota koeficientu  $a$  je:  $a = 0.01$ .



Obrázek 2.8: Architektura sítě YOLOv3.

## Trénování modelu YOLO

Co se týče učení modelu YOLO, je třeba definovat ztrátovou funkci modelů. V dnešní době neexistuje univerzální ztrátová funkce, je tedy třeba při výběru zvážit čas konvergence, typ řešeného úkolu a další aspekty. YOLOv3 využívá L2 ztrátovou funkci, která se snaží minimalizovat součet kvadrátů rozdílů mezi anotacemi a predikcemi modelu a je dána vztahem [37, 39]:

$$L_2(\hat{y}, y) = \sum_{i=0}^m (y^i - \hat{y}^i)^2, \quad (2.5)$$

kde  $y^i$  reprezentuje anotace a  $\hat{y}$  predikce modelu.  $m$  pak odpovídá celkovému počtu vzorků.

Ztrátová funkce  $L_2$  využívá metriku IoU, která je dána vztahem:

$$IoU = \frac{A \cap B}{A \cup B}, \quad (2.6)$$

Jelikož každá buňka rozdělení vstupního obrázku zahrnuje několik ohraničujících rámečků, je třeba vybrat ohraničující rámeček „zodpovědný“ za daný objekt. Kvůli tomu je vybrán rámeček s nejvyšší hodnotou IoU predikce a anotace. Loss funkce YOLOv3 se skládá ze tří částí, klasifikační ztrátové funkce, lokalizační ztrátové funkce a skórovací ztrátové funkce [37, 39]. Klasifikační ztrátová funkce je dána vztahem:

$$\sum_{i=0}^{S^2} \mathbb{I}_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2, \quad (2.7)$$

kde  $\mathbb{1}_i^{obj} = 1$  pokud se objekt nachází v buňce  $i$ , jinak 0 a  $\hat{p}_i(c)$  představuje podmíněnou pravděpodobnost třídy  $c$  pro buňku  $i$ .

Lokalizační ztrátová funkce představuje chybu v umístění a velikosti predikovaného ohraničujícího rámečku oproti rámečku anotace.

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2], \quad (2.8)$$

kde  $\mathbb{1}_{ij}^{obj} = 1$  pokud  $j$ -tý ohraničující rámeček v buňce  $i$  zodpovídá za detekci objektu, jinak 0.  $\lambda_{coord}$  zvyšuje váhu lokalizační váhové funkce. Chyba umístění a velikosti není stejně významná pro velké a malé ohraničující rámečky (například rozdíl 2 pixelu není stejně významný). Tento fakt je v rámci ztrátové funkce částečně reflektován pomocí druhé mocniny šířky a výšky rámečků [37, 39].

Poslední částí ztrátové funkce YOLOv3 je skórovací ztrátová funkce, která je pro případy, že je v rámečku objekt detekován, dána vztahem:

$$\sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} (C_i - \hat{C}_i)^2, \quad (2.9)$$

kde  $\hat{C}_i$  je skóre pro rámeček  $j$  v rámci buňky  $i$  a  $\mathbb{1}_{ij}^{obj} = 1$  pokud  $j$ -tý rámeček v buňce  $i$  je zodpověď za detekci objektu, jinak 0.

Pokud objekt detekován není, je skórovací ztrátová funkce dána vztahem:

$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2, \quad (2.10)$$

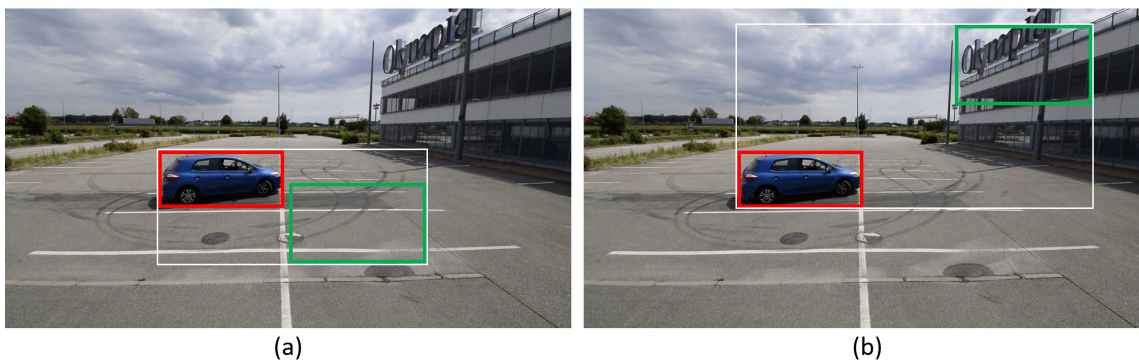
kde  $\mathbb{1}_{ij}^{noobj}$  je doplněk  $\mathbb{1}_{ij}^{obj}$ ,  $\hat{C}_i$  je skóre pro rámeček  $j$  v rámci buňky  $i$  a  $\lambda_{noobj}$  slouží pro snížení významu chyby při detekci pozadí. Počet buněk, ve kterých se nějaký objekt nachází je menší než počet buněk, ve kterých jsou objekty detekovány, což způsobuje problém nevyváženosti rozpoznávaných tříd (objekt a ne-objekt) [37, 39].  $\lambda_{noobj}$  slouží pro správné zahrnutí této skutečnosti do ztrátové funkce. Celková ztrátová funkce YOLOv3 je pak dána jako součet jednotlivých ztrátových funkcí.

Vylepšení ztrátové funkce YOLOv3 může spočívat ku příkladu v použití GIoU místo IoU. GIoU je dána vztahem [43]:

$$GIoU = IoU - \frac{C \setminus (A \cap B)}{|C|}, \quad (2.11)$$

kde  $C$  reprezentuje nejmenší ohraničující obdelník rámečků  $A$  a  $B$ , na obrázku 2.9 označen bílým obdelníkem. Výhodou použití GIoU je lepší vyjádření podobnosti ohraničujících rámečků v situacích, kdy se neprotínají. IoU má v těchto situacích nulovou hodnotu zatímco GIoU nikoliv [43]. Rozdíl mezi IoU a GIoU je možné určit na základě obrázku 2.9.





Obrázek 2.9: Příklady pro ukázkou rozdílu mezi GIoU a IoU.

Hodnota IoU pro obrázky (a) i (b) (obrázek 2.9) je nulová. Hodnota GIoU pro obrázek (a) však nabývá větší hodnoty než pro obrázek (b). Obecně pro rámečky, které se nepřekrývají platí, že hodnota GIoU je větší, čím je vzdálenost mezi rámečky menší. GIoU tedy pro účely ztrátové funkce řeší problém úpravy parametrů trénování modelu v případech, kdy se predikovaný rámeček nepřekrývá s rámečkem anotace.

## Vyhodnocení modelu YOLO

Pro potřeby evaluace detekčních systémů je třeba stanovit vhodné metriky. Nejpoužívanější metrikou, která se pro tyto typy systémů využívá je „mean average precision“. Ta je dána jako aritmetický průměr interpolovaných hodnot „average precision“ (AP) pro každou třídu pro danou množinu detekcí [20]. AP vyhodnocené pro každou detekovanou třídu je pak dáno jako plocha pod „precision“/„recall“ (PR) funkcí. PR je pak definována přiřazením každé z detekcí k anotaci, se kterou nabývá detekce největší hodnoty IoU [20]. Pokud se detekce a anotace překrývají z více než například 50%, je detekce považována za „true positive“ pro danou anotaci a všechny ostatní detekce jako „false positive“ [20]. Dále jsou pak spočteny hodnoty „recall“ a „precision“ pro podmnožiny detekcí seřazených podle jejich skóre v sestupném pořadí. „recall“ je definováno jako poměr anotací a „true positive“ detekcí. „precision“ pak udává poměr mezi „true positive“ detekcemi a všemi detekcemi [20].

## Kapitola 3

# Současný stav fúzních algoritmů

Tato kapitola se zaměřuje na, pro tuto práci relevantní, fúzní algoritmy, které využívají senzorická data z milimetrového radaru a kamery pro detekci objektů. Nejedná se tedy o encyklopedický přehled, nýbrž o přehled metod důležitých pro tuto práci kvůli maximálnímu celkovému rozsahu této práce. Mnoho těchto zkoumaných fúzních algoritmů se zaměřuje na detekci překážek pro účely řízení vozidel a senzory jsou v těchto případech typicky umístěny na automobilech. Některé tyto algoritmy lze však použít i pro monitorování provozu a jiné případy použití. Dále jsou v této kapitole rozebrány možné reprezentace signálů milimetrového radaru pro účely popsaných fúzních metod. Závěrem se věnuje synchronizačním a kalibračním postupům. Zaměřuje se kvůli maximálnímu rozsahu práce pouze na postupy a prostředky, které jsou vhodné pro synchronizaci a kalibraci milimetrového radaru a kamery.

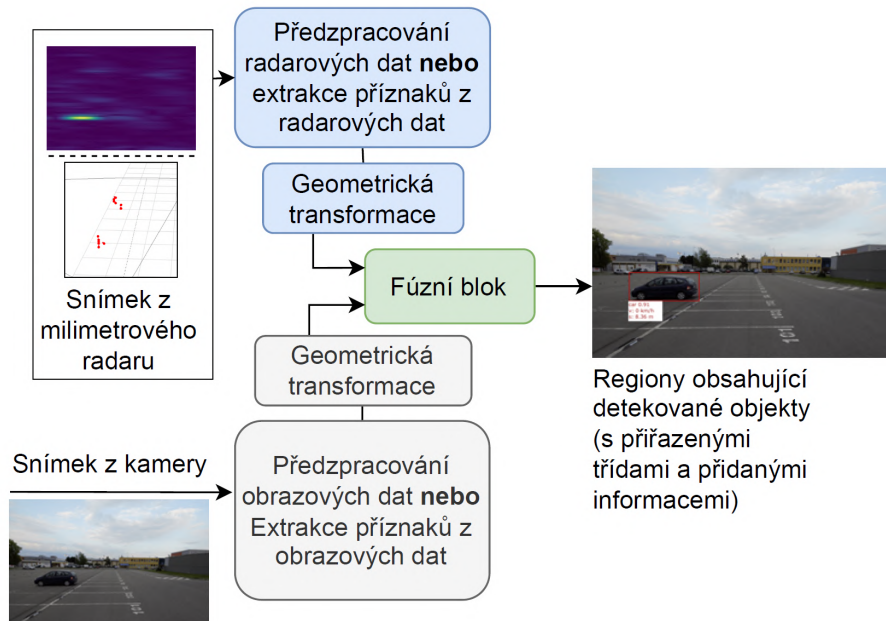
### 3.1 Dělení fúzních algoritmů

Pro pochopení různých fúzních algoritmů je žádoucí rozlišit různé způsoby využití senzorických dat. Tato část se tedy zaměřuje na identifikaci způsobů fúze senzorických dat z milimetrového radaru a kamery.

#### Rozdělení fúzních algoritmů podle způsobu použití dat

Za základní rozdělení fúzních algoritmů lze považovat dělení podle způsobu použití dat z obou sensorů. Tímto způsobem lze rozlišit algoritmy, které využívají data z kamery a radaru jak pro trénování systému, tak při nasazení, a algoritmy, které používají data z kamery pouze pro trénování a při nasazení spoléhají pouze na data z milimetrového radaru. Oba tyto způsoby fúze mají své výhody a využití, jenž jsou dále rozebrány.

Systémy, které využívají data z obou sensorů při nasazení lze reprezentovat obrázkem 3.1. Podle úrovně fúze jsou pak některé bloky důležitější než jiné. Pokud například dochází k brzké fúzi dat, bloky zajišťující předzpracování a extrakci příznaků budou méně důležité, než v případě pozdní fúze (více viz v části 3.1). Na obrázku jsou dále vidět dvě často používané reprezentace dat z milimetrového radaru (mračno bodů a vzdálenostně-úhlová mapa), které mohou být použity (více viz v části 3.2). Výstupem takovýchto systémů mohou být regiony obsahující detekované objekty v souřadném systému kamery (jako na obrázku), nebo případně označení objektů v souřadném systému radaru.



Obrázek 3.1: Koncepce systémů využívajících data z kamery i milimetrového radaru pro trénování i nasazení.

Mezi výhody systémů využívajících data z obou senzorů při nasazení lze zahrnout fakt, že kamera a milimetrový radar lze označit jako komplementární senzory. Je tedy možné získat fúzi dat z obou senzorů robustnější systém, který díky milimetrovému radaru pracuje i ve zhoršených podmínkách jako déšť, či mlha, více viz v části 4.

Do kategorie systémů pracujících s daty z obou senzorů patří systém RVNet [24] zaměřující se jak na detekci překážek, tak na rozdělení objektů do tříd: vozidla, chodci, jednostopá vozidla a jiné objekty. RVNet pracuje se snímky z kamery a odpovídajícími mračny bodů kalibrovanými do souřadného systému kamery. RVNet využívá části architektury konvoluční neuronové sítě YOLO [39] pro získání příznaků z obrazových dat a výstupní vrstvu pro detekci a klasifikaci.

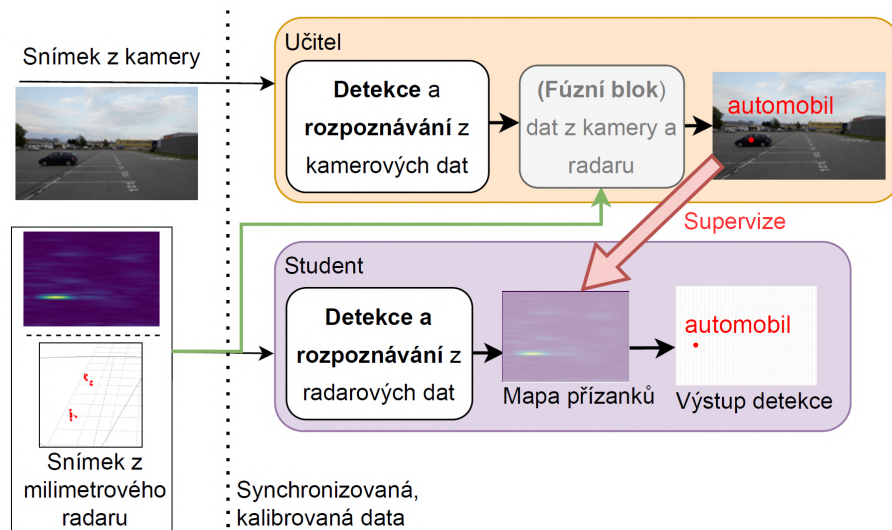
Dalším příkladem je systém vyvinutý skupinou vedenou L.Teck-Yianetem [27]. Tento systém data z radaru a kamery zpracovává samostatně a následně fúzuje výsledné prostorové příznakové mapy (spatial feature maps). Kalibrace dat 3.5 je provedena na začátku zpracování, aby byly příznakové mapy z dat z radaru a z kamery zarovnané. Na rozdíl od RVNet [24] jsou pro reprezentaci radarových signálů použity 2D vzdálenostě-úhlové mapy (range-azimuth maps), které oproti mračnům bodů nezpůsobují ztrátu kontextuálních informací z radarových signálů (více viz v části 3.2). Prostorové příznakové mapy jsou škálovány a použity jako vstup SSD neuronové sítě [30].

Systémy, které využívají data z kamery pouze pro trénování, mají oproti systémům využívajícím kameru, výhodu v místech, kde není možné umístit kameru nebo v případech, kdy senzor pracuje ve špatných světelných podmínkách, nepříznivém počasí nebo v situacích, kdy jsou při měření přítomny poletující částice. Jak je popsáno v části 4, milimetrovému radaru nečiní problém pracovat v nepříznivých podmínkách. Pokud je tedy možné získat systém, který pracuje pouze s daty z radaru, bude robustnější než systém založený pouze na datech z kamery.

Nevýhodou tohoto přístupu je však nízká významnost příznaků, které lze získat z radarových dat pro rozpoznávání různých objektů. Většina systémů, které při nasazení využívají

pouze data z radaru, je tedy zaměřena pouze na detekci překážek dále neklasifikovaných překážek. Tyto systémy tedy nejsou schopné rozpoznat ku příkladu člověka od cyklisty. Existují však i takové systémy, které na základě radarových signálů provádí kromě detekce i klasifikaci do základních tříd. Mezi tyto systémy patří například RODNet [51], který za pomoci vzdálenostně-úhlových map rozlišuje vozidla, cyklisty a chodce. Pro trénování využívá fúzi dat z radaru a kamery.

Následující obrázek 3.2 reprezentuje základní koncepci systémů, které data z kamery využívají pro trénování. Snímek z kamery je v části „učitel“ použit pro detekci, tím jsou získány umístění objektů a jejich třídy. V rámci RODNet [51] následuje fúzní blok, ve kterém se kombinují získané anotace z dat z kamery s informacemi o umístění získanými z radarových dat pomocí CFAR detektoru [51]. Anotace získané buď pouze z obrazových dat bez fúzního bloku, nebo fúzí anotací z obrazových dat a radarových dat, jsou použity pro trénování části „student“. Část „student“ pak může být použita samostatně a využívat data pouze z milimetrového radaru.



Obrázek 3.2: Koncepce systémů využívajících data z kamery pouze pro trénování.

## Rozdělení fúzních algoritmů podle úrovně fúze

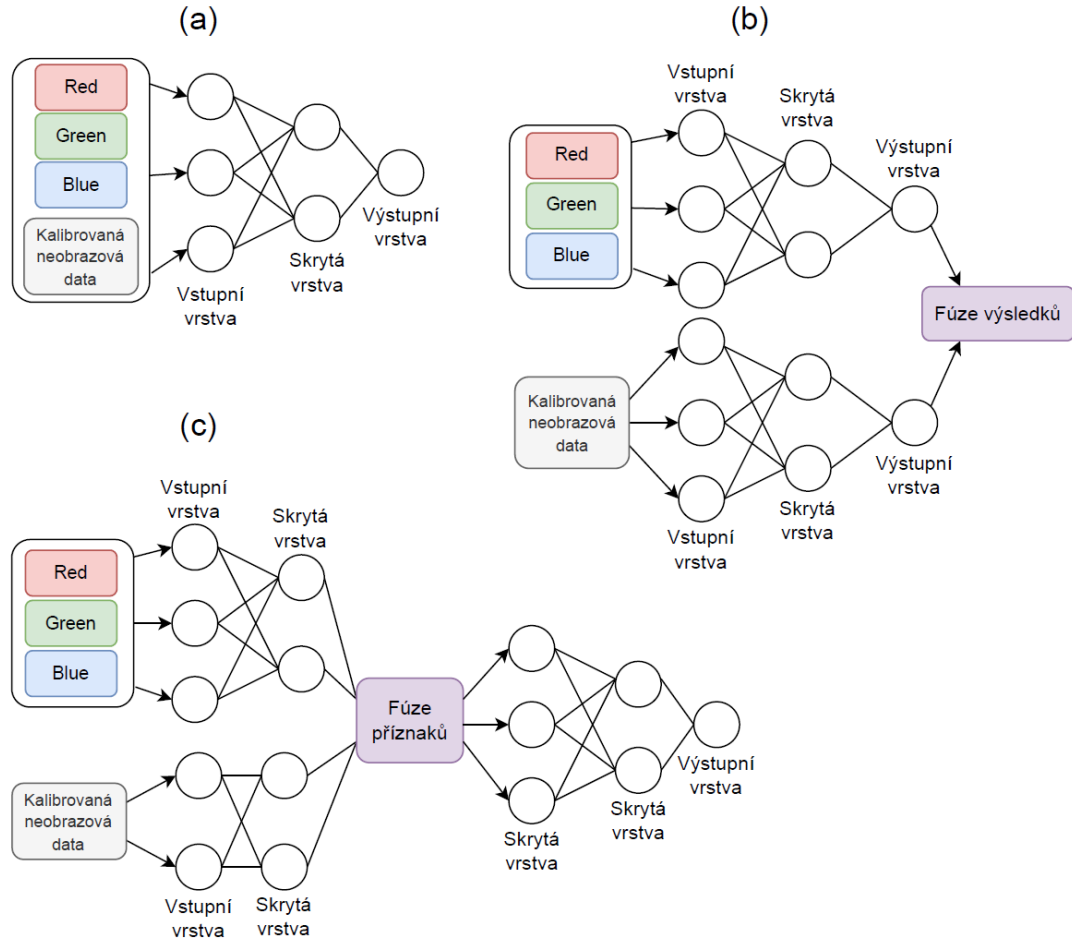
Dalším důležitým aspektem fúzních algoritmů je úroveň fúze. Typicky jsou rozlišovány čtyři základní úrovně fúze dat, a to fúze na úrovni pixelů, brzká fúze, pozdní fúze a průběžná fúze [6]. Úrovně fúze jsou ilustrovány pomocí architektur neuronových sítí na obrázku 3.3.

Pro metody hlubokého učení je často používána brzká fúze, kdy jsou neobrazová data upravena tak, aby mohla být konkatenována s obrazovými daty a použita jako vstup neuronové sítě. Z 3-kanálových obrazových dat, tedy přidáním neobrazových dat vzniká 4-kanálový vstup neuronové sítě. Výhodou tohoto přístupu je schopnost neuronové sítě naučit se vztahy mezi zdroji vstupních dat (senzory) [6].

Průběžná fúze je také často používána, obzvlášť pro kombinaci milimetrového radaru a kamery. Zástupcem tohoto druhu fúze je ku příkladu RVNet [24] zmíněný výše. Průběžná fúze se vyznačuje tím, že je použita pouze jedna neuronová síť s několika oddělenými vstupy pro každý senzor [6]. V rámci architektury neuronové sítě pak dojde k fúzi příznaků získaných z dat z jednotlivých senzorů. V rámci RVNet jsou data z milimetrového radaru

a kamery zpracovány symetrickými vrstvami odděleně, vzniklé příznaky jsou konkaténovány a dále použity společně ve zbylých vrstvách neuronové sítě pro detekci a klasifikaci [24].

Pozdní fúze spočívá v použití několika nezávislých systémů, které extrahují příznaky z dat získaných pomocí různých senzorů [6]. Příznaky jsou pak fúzovány až na závěr za pomoci probablistických technik [49].



Obrázek 3.3: Rozdělení fúzních algoritmů podle úrovně fúze: (a) brzká fúze, (b) pozdní fúze, (c) průběžná fúze.

## Přehled dalších fúzních metod

Tato část se zabývá souhrnem konkrétních fúzních metod. Některé již byly zmíněny a popsány výše. Jak je zmíněno na začátku této kapitoly, většina fúzních metod se zaměřuje na detekci překážek pro účely řízení vozidel. Mezi metody, které rozeznávají pouze dvě třídy: překážka a volný prostor, patří ku příkladu práce SO-Net: Joint Semantic Segmentation and Obstacle Detection Using Deep Fusion of Monocular Camera and Radar [23] nebo RRPN: Radar Region Proposal Network for Object Detection in Autonomous Vehicles [32].

Mezi metody detekující více tříd patří Spatial Attention Fusion for Obstacle Detection using MmWave Radar and Vision Sensor [10] nebo výše zmíněné metody RODNet [51] a RVNet [24].

Metoda SO-Net se řadí mezi metody průběžné fúze dat a stejně jako RVNet [24] využívá části architektury modelu YOLO [39]. Tato práce má však několik potenciálních chyb jako tendence sítě k přetrénování a dále, že není v práci zvaženo, že mračno bodů z milimetrového radaru je oproti mračnu bodů z ku příkladu lidarů řídké. Autoři však ukázali, že jejich přístup je lepší, než jiné fúzní metody, které v práci zvažili [5].

Další metoda, která je zaměřena na detekci překážek je RRPN [32]. Ta využívá modelu Fast-R-CNN [15], kdy jsou pro generování návrhů částí obrázku (RPN), kde se s vysokou pravděpodobností nachází objekty, využívána do souřadného systému kamery kalibrovaná radarová data. Tento proces dokáže urychlit a zpřesnit proces generování návrhů pro klasifikační část sítě [32] (více viz v části 2.3).

Poslední zde zmíněnou metodou je práce Shuo Chunga a dalších [10]. Ta využívá „spatial attention“ fúzi (SAF) a jedná se tedy o metodu typu průběžné fúze. Autoři navrhnou fúzní blok skládající se z konvolučních vrstev, který využívá obrazovou reprezentaci dat z radaru pro generování „spatial attention“ informace, která je dále fúzována s příznaky generovanými pomocí obrazových dat [10].

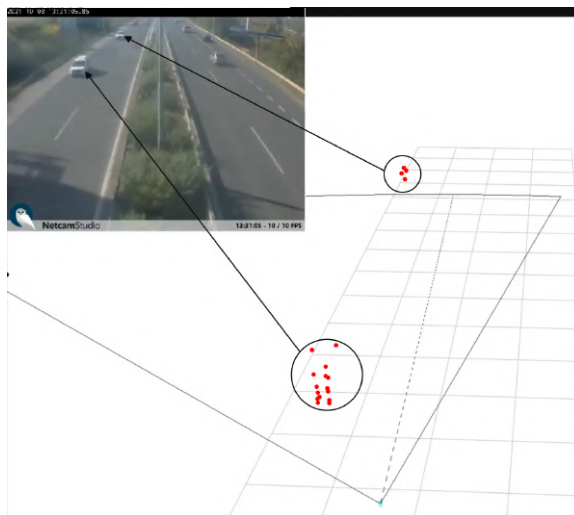
## 3.2 Způsoby reprezentace signálů milimetrového radaru

Aby bylo možné využít signály z milimetrového radaru pro účely výše popsaných fúzních metod, je třeba je interpretovat. Tato část popisuje různé způsoby reprezentace signálů z milimetrového radaru. Mezi reprezentace signálů z milimetrového radaru patří vzdálenostně-úhlové mapy, mračna bodů nebo mřížky obsazenosti.

### Reprezentace dat pomocí mračna bodů

Mračno bodů je často používaná reprezentace signálů z milimetrového radaru. Mnoho jiných senzorů, jako lidar, produkuje mračna bodů. Tyto metody mohou být určeny pro segmentaci, klasifikaci, či detekci a mohou být adaptovány pro mračna bodů pocházejících z milimetrového radaru. Mezi tyto metody primárně určené pro lidar patří PointNet++ zaměřující se na segmentaci mračna bodů. V práci Danzera a dalších [11] je architektura systému PointNet++ upravena tak, aby dokázala pracovat s mračny bodů generovanými právě milimetrovým radarem. Důvodem, proč je třeba metody určené pro lidar adaptovat pro potřeby milimetrového radaru je, že mračna bodů pocházejících z radaru jsou řídká kvůli způsobu získávání, kdy je mračno bodů získáno prahováním výstupu Fourierovy transformace signálů z radaru (více viz v kapitole 4).

Příklad mračna bodů pocházejícího z milimetrového radaru je ukázán na obrázku 3.4. Jednotlivé shluky bodů reprezentují automobily na odpovídajícím snímku a kužel pak zorné pole radaru s čárkovanou čarou vyobrazující dělení směrů provozu. Každý z automobilů je reprezentován pouze několika body, je zde tedy možné pozorovat rozdíl hustoty mračna bodů oproti mračnu bodů získaného z lidarů.



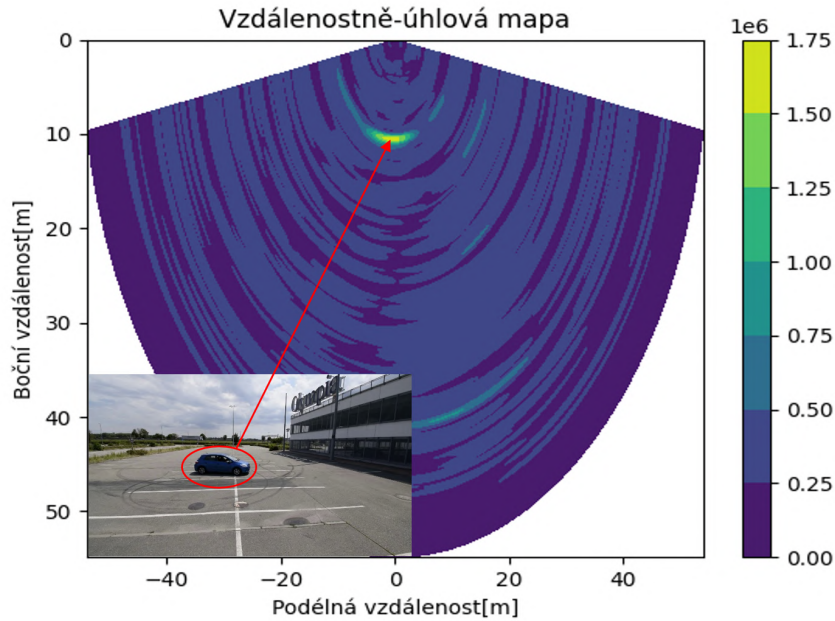
Obrázek 3.4: Příklad mračna bodů získaného z milimetrového radaru.

### Reprezentace dat vzdálenostně-úhlovými mapami

Vzdálenostně-úhlové mapy případně vzdálenostně-rychlostně-úhlové mapy jsou stejně jako mračna bodů vytvořena pomocí Fourierovy transformace radarových signálů<sup>1</sup>. V případě této reprezentace však nedochází k prahování, a tím ke ztrátě informace. Místo prahování je použitý celý výstup Fourierovy transformace, a tak zachována většina dostupných informací získaných radarem.

Na obrázku 3.5 je ukázka teplotní mapy vyobrazující vzdálenostně-úhlovou mapu společně s odpovídajícím snímkem z kamery. Hodnoty teplotní mapy vyobrazují intenzitu odrazu od objektů ve scéně. Žlutě označený vrchol odpovídá odrazu automobilu zaznamenaného milimetrovým radarem. Popis zpracování signálu z milimetrového radaru a vytváření vzdálenostně-úhlové mapy jsou dostupné v části 4.

<sup>1</sup>Fourierova transformace slouží k převodu odraženého signálu z časové domény do frekvenční a umožňuje stanovit vzdálenost, rychlost a úhel objektů ve scéně. Více informací v části 4.



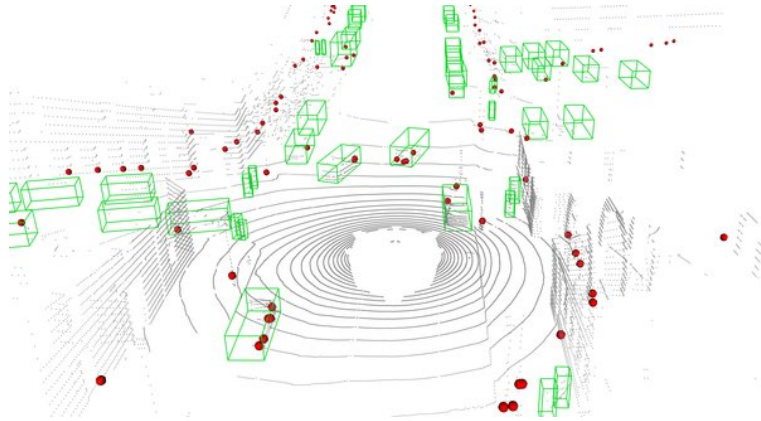
Obrázek 3.5: Příklad vzdálenostně-úhlové reprezentace z milimetrového radaru.

### 3.3 Datové sady pro fúzi informací z milimetrového radaru a kamery

Datové sady obsahující data z milimetrového radaru a kamery jsou zaměřeny na mnoho různých typů scén a případů užití. Liší se dále reprezentací radarových dat a v dalších aspektech.

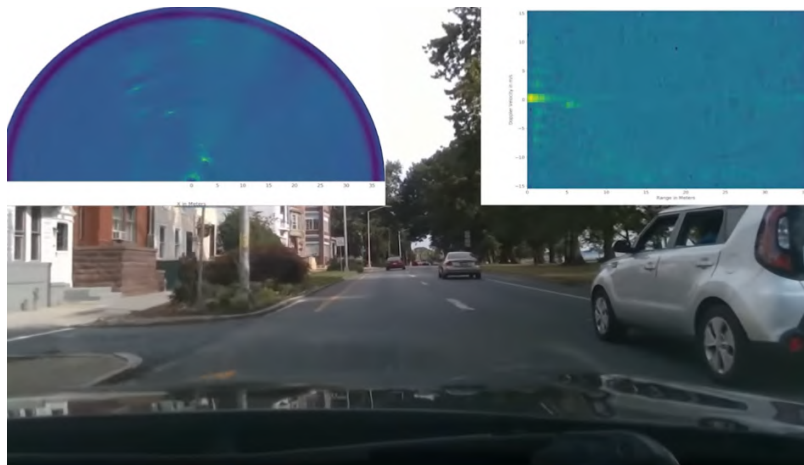
Nejoblíbenější datovou sadou využívanou pro fúzní metody radaru a kamery je Nuscenes [9]. Ta je používána v mnoha systémech shrnutých v rámci přehledové studie [5]. Nuscenes obsahuje data z lidarů, pěti milimetrových radarů a šesti kamer rozmístěných tak, aby pokrývali 360° zorné pole [9]. Data se skládají z 1000 scén dlouhých 20 s, které jsou plně anotovány pomocí 3D ohraničujících rámečků s 23 třídami a 8 atributy (ku příkladu aktivita, póza, viditelnost) [9]. Data z milimetrových radarů jsou však u této datové sady reprezentována pouze mračnem bodů, není tedy možné použít Nuscenes datovou sadu, pokud je třeba použít jinou reprezentaci jako vzdálenostně-úhlovou mapu a jiné. Na obrázku 3.7 je vyobrazen příklad snímku z datové sady Nuscenes.





Obrázek 3.6: Ukázka snímku z datové sady Nuscenes, červené body odpovídají mračnu bodů milimetrového radaru, anotace označené zelenými 3D ohraničujícími rámečky a mračnem bodů z lidarů (šedě)<sup>3</sup>.

Další datová sada, která se zaměřuje na fúzi milimetrového radaru a kamery je datová sada RaDICAL [28]. Jedná se o datovou sadu, která vznikla na konci roku 2021 a oproti datové sadě Nuscenes jsou zde data z milimetrového radaru reprezentována na nejnižší možné úrovni před dalším zpracováním (více viz v kapitole 4). Tím je možné při použití této datové sady reprezentovat data milimetrového radaru libovolným způsobem 3.2. Uživatel tedy není odstíněn od parametrů milimetrového radaru a musí používat data, která byla snímána s určitým nastavením milimetrového radaru v daných podmínkách. RaDICAL zahrnuje data sesbírána jak ve vnitřním, tak ve vnějším prostředí. Dále pak jsou zde rozložena data snímána ve venkovním prostředí s nastavením s maximálním dosahem 30 m, 50 m.



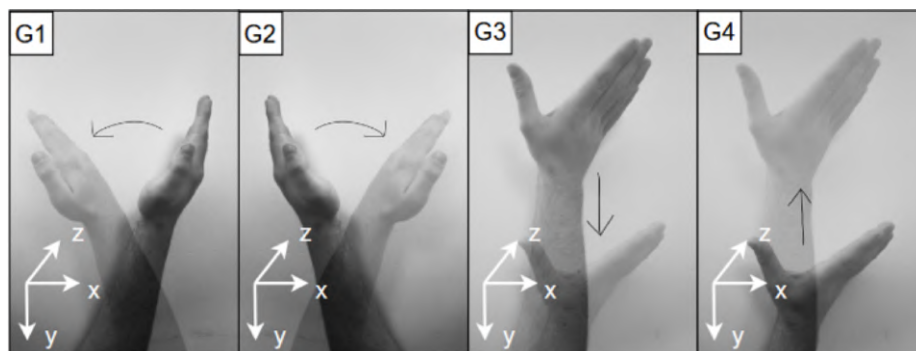
Obrázek 3.7: Ukázka snímku z datové sady RaDICAL vyobrazující snímky kamery, vzdálenostně-úhlovou mapu a vzdálenostně-rychlostní mapu<sup>5</sup>

Mezi datové sady obsahující data z FMCW milimetrového radaru a kamery náleží specifická datová sada, která se zaměřuje na gesta rukou Hand Gestures Recorded with

<sup>3</sup>Obrázek byl převzat z práce [9].

<sup>5</sup>Obrázek byl převzat z práce [28].

MMWAVE FMCW Radar (AWR1642) [29]. Tato datová sada zahrnuje 4600 vzorků s 12 různými pohyby ruky a jsou sesbírány pro čtyři různé osoby. Data jsou zde reprezentována pomocí rychlosti, vzdálenosti, souřadnic a maximální energie odrazu. Jedná se tedy o příklad úzce zaměřené datové sady, která ukazuje univerzálnost milimetrového radaru. Ukázkou rozpoznávaných gest z této datové sady je obrázek 3.8. Mezi další datovou sadu s takto úzkým zaměřením lze zařadit ku příkladu datovou sadu prezentovanou v rámci práce One-shot learning-based driver's head movement identification using a millimetre-wave radar sensor [33], která se zaměřuje na sledování pohybu hlavy pro účely určení úrovně pozornosti řidiče automobilu.



Obrázek 3.8: Ukázky rozpoznávaných pohybů z datové sady Hand Gestures Recorded with MMWAVE FMCW Radar (AWR1642)<sup>7</sup>.

### 3.4 Postupy časové synchronizace senzorických dat

Synchronizace senzorů je při snaze o fúzi senzorických dat kritická. Bez přesné synchronizace by nebylo ve většině případů z fúze senzorických dat možné benefitovat. Časová synchronizace senzorů je pak dosažitelná různými způsoby. Tato část popisuje aktuálně dostupnou hardware a metody. Dále se zaměřuje na kompatibilitu těchto postupů s použitím kamery a milimetrového radaru.

#### Synchronizace pomocí síťových protokolů

Jako první možný způsob časové synchronizace, je použití zařízení podporujících protokoly navržené pro přesnou časovou synchronizaci. Příkladem takovýchto protokolů jsou Precision Time Protocol (PTP) [1] a Network Time Protocol (NTP) [31].

#### Precision Time Protocol

Precision Time Protocol (PTP) je síťový protokol navržený pro synchronizaci zařízení připojených v rámci sítě Ethernet a zaručuje vysoký stupeň synchronizace s časovou nejistotou menší než  $1\mu$  [1]. Časová synchronizace je v případě PTP protokolu zabezpečena zasláním speciálních zpráv s časovými značkami mezi zařízeními v rámci jedné sítě. V inicializační fázi je vybráno zařízení s nejpřesnějšími hodinami (clock), které je určeno jako „master“ zařízení. Ostatní připojená zařízení zaujmou roli „slave“ zařízení [1]. Díky tomuto rozdělení

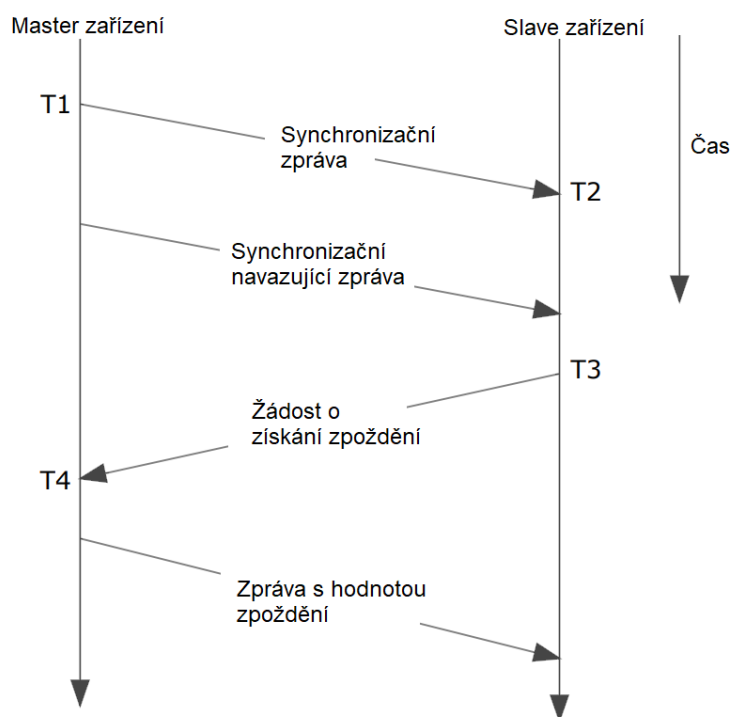
<sup>7</sup>Obrázek byl převzat z práce [29]

je pak možné za pomoci zpráv určit odchylku hodin zařízení od hodin „master“ zařízení i dobu zpoždění zprávy danou přenosem zpráv po síti.

Komunikace mezi „master“ a „slave“ zařízeními je vyobrazena na obrázku 3.9, kde T1 a T4 označují přesný čas odeslání a přijetí zprávy „master“ zařízením, T2 a T3 jsou přesné časy přijetí a odeslání zprávy „slave“ zařízením. Offset  $T_{diff}$  mezi hodinami „slave“ a „master“ zařízení je pak vyjádřen jako [1]:

$$T_{diff} = \frac{(T2 - T1) - (T4 - T3)}{2},$$

Zařízení „slave“ poté využívá hodnotu  $T_{diff}$  pro úpravu času svých hodin tak, aby odpovídala času hodin zařízení „master“. Tento postup předpokládá konstantní zpoždění přenosu mezi zařízeními a může být zpřesněn použitím algoritmu pro kompenzaci času přenosu po síti podle úrovně zatížení nebo algoritmu kompenzující nepřesnost hodin „slave“ zařízení způsobenou změnou teploty [1].



Obrázek 3.9: PTP komunikace mezi master a slave zařízením.

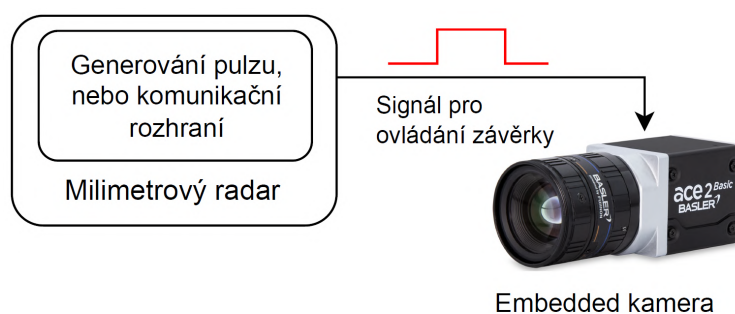
### Network Time Protocol

Network Time Protocol (NTP) je síťový protokol pro synchronizaci času zařízení v rámci sítě stejně jako PTP. NTP zajišťuje synchronizaci s časovou nejistotou v řádu milisekund (místo časové nejistoty PTP: do  $1\mu$ ), na rozdíl od PTP protokolu totiž nebere v potaz zpoždění způsobené přenosem po síti mezi zařízeními. NTP protokol není pro synchronizaci senzorů pro účely fúze typicky používán [31].

## Synchronizace pomocí přímé signalizace

Tento postup předpokládá použití embedded zařízení za účelem ovládní snímání daného senzoru. Možným příkladem může být milimetrový radar a embedded kamera, kdy milimetrový radar vysílá signály pro ovládní závěrky kamery. Tím je možné při zahájení snímání nového snímku radarem vyslat signál kameře, která zahájí expozici, a tím budou snímky z obou senzorů přesně synchronizovány. Tento příklad je ilustrován na obrázku 3.10 a varianta popsaného přístupu je navržena v práci Yong Zhoua ad. [53].

Dalším příkladem použití embedded kamer pro účely časové synchronizace může být synchronizace několika kamer připojených k externímu zařízení a následné využití časových značek kamer a externího zařízení k přesné synchronizaci. Varianta tohoto přístupu je navržena v práci Hyunse Yoona a dalších [52] pro získání 3D modelu za použití několika kamer.



Obrázek 3.10: Příklad použití embedded kamery a milimetrového radaru pro časovou synchronizaci ovládním závěrky kamery radarem.

## Časová synchronizace na základě příznaků z dat senzorů

Dalším možným postupem pro časovou synchronizaci je využít příznaky získané z dat ze senzorů [13, 12, 35]. Použitelnými příznaky mohou být video, audio, akcelerometr, či telemetrie. Zajímavým automatizovaným synchronizačním postupem je práce Lex Fridmana ad. [13], kdy jsou pro synchronizaci několika kamer umístěných ve vozidle použity vibrace způsobené pohybem automobilu. Tyto vibrace pak mohou sloužit k synchronizaci pomocí vzájemné korelace signálů z jednotlivých kamer. V této práci bylo docíleno průměrné přesnosti synchronizace kamer do 13,5ms.

Dalším příkladem použití příznaků pro synchronizaci senzorů je práce A. Elhayeka [12]. V této práci autoři využívají pro nalezení odpovídajících snímků z kamer trajektorie pohybu, které jsou dále filtrovány RANSAC algoritmem [12]. Tento přístup využívají pro synchronizaci videí, ve kterých se nachází rychle se pohybující lidé, především akrobati a parkouristé. Navržený systém dosahuje průměrné přesnosti určení snímků oproti referenčním datům: 0,75 snímku. Výhodou této metody je, jeho použitelnost pro kamery s různou snímkovou frekvencí.

## Jiné způsoby časové synchronizace

Mezi alternativní způsoby synchronizace patří použití filmové klapky pro identifikaci zahájení snímání, nebo umístění ukazatele času do obrazu [5]. Ani jeden z těchto způsobů však není možné použít pro kombinaci senzorů produkujících obrazová a neobrazová data,

způsobují částečné znehodnocení záznamu a je obtížné tyto synchronizační postupy použít pro automatickou synchronizaci [5].

### 3.5 Způsoby kalibrace milimetrového radaru a kamery

Tato část zkoumá možnosti kalibrace milimetrového radaru a kamery. Jak je popsáno v části 4 a 3.2, data z milimetrového radaru jsou po zpracování signálů v jednotkách reálného světa, což pro monokulární kameru neplatí. Je tedy třeba nalézt způsob, jak data získaná milimetrovým radarem transformovat, aby je bylo možné zobrazit v souřadném systému kamery.

#### Kalibrace pomocí vnitřních a vnějších parametrů kamery

Za předpokladu, že soustava souřadnic milimetrového radaru odpovídá souřadnému systému reálného světa, je možné transformaci mezi kamerou a milimetrovým radarem realizovat pomocí nalezení vnitřních a vnějších parametrů kamery. Parametry kamery je poté možné využít k transformaci ze souřadného systému světa (radaru) do souřadného systému kamery následovně:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} [R|t] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}, \quad (3.1)$$

kde souřadnice  $(x_w, y_w, z_w)$  jsou souřadnice reálného světa získané z milimetrového radaru a  $(u, v, 1)$  jsou odpovídající souřadnice v souřadném systému průmětny kamery. Pokud se jedná o 2D mračno bodů, kdy není známá informace o výškovém umístění detekovaných objektů, je třeba dále předpokládat, že se všechny body nachází v určité výšce. K tomuto účelu může sloužit ku příkladu výška, v jaké se nachází radar, všechny  $z_w$  pak budou nabývat hodnotu rovnající se výšce umístění radaru. Matice rotace  $R$  ( $3 \times 3$ ) a vektor translace  $t$  pak definují vnější parametry vyjádřením pozice souřadných soustav světa v systému souřadnic kamery. Parametry  $f_x, f_y$  odpovídají ohniskovým vzdálenostem a  $c_x, c_y$  jsou souřadnice určující hlavní bod [40].

Zjištění vnějších parametrů transformace je v případě kalibrace milimetrového radaru a kamery snadné. Je si však třeba uvědomit, že jak translační vektor  $t$ , tak rotační matice  $R$  vyjadřují pozici souřadných soustav světa v systému souřadnic kamery. Pokud je tedy známa výška kamery nad zemí a směr kamery v prostoru, je třeba matici  $[R|t]$  invertovat [40].

Pro nalezení vnitřních parametrů kamery je v případě popisovaného postupu kalibrace možné využít fotografií šachovnice v různých pozicích před objektivem kamery. Nalezení vnitřních parametrů kamery je pak realizovatelné ku příkladu pomocí Python knihovny OpenCV2, která obsahuje potřebné metody.

Varianta popsaného způsobu kalibrace je využívána pro kalibraci milimetrového radaru a kamery v práci Radar and vision sensors calibration for outdoor 3D reconstruction [14], která se zaměřuje na kalibraci těchto senzorů pro účely 3D rekonstrukce scény. Dalším příkladem využití tohoto kalibračního postupu lze nalézt v předchozí práci [40], na kterou je zde volně navázáno.

Použití kalibrace spoléhající se na vnitřní a vnější parametry kamery přináší potenciální problém spočívající v získání souřadnic z milimetrového radaru, které se liší od reálných

souřadnic světa. Tento problém by bylo možné kompenzovat použitím transformace ze souřadného systému radaru do souřadného systému světa a poté využitím výše popsaného kalibračního postupu. Tato forma kalibrace je použita například v diplomové práci Kalibrace robotického pracoviště [50], kde je popsaným postupem kalibrováno robotické rameno a kamera.

## Direct Linear Transform (DLT)

V rámci práce Jiyong Oha [34] jsou porovnány různé kalibrační metody vhodné pro kalibraci milimetrového radaru a kamery. Jedním z nich je metoda Direct Linear Transform, která dokáže transformovat souřadnice získané z milimetrového radaru přímo do souřadného systému kamery. Tyto souřadnice mohou být jak dvoudimenzionální, tak třídimenzionální. Pokud milimetrový radar produkuje data pouze v rovině bez informace o elevaci, je třeba použít 2D DLT, pokud je výstupem radaru i informace o výšce nad zemí, je třeba využít 3D DLT.

### Algoritmus 2D DLT

V této verzi algoritmu je prováděna projekce mezi dvěma rovinami: souřadným systémem radaru ( $p_i = (x_i, y_i)$ ) a systémem souřadnic kamery ( $q_i = (u_i, v_i)$ ). Výhodou této metody je schopnost nalezení lineárního zobrazení mezi dvěma množinami.

$$\tilde{\mathbf{q}} = \mathbf{H}\tilde{\mathbf{p}}, \quad (3.2)$$

kde  $\tilde{\mathbf{q}}$  a  $\tilde{\mathbf{p}}$  jsou homogenní souřadnice  $\mathbf{q}$  a  $\mathbf{p}$  a  $\mathbf{H}$  odpovídá následující matici:

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (3.3)$$

Po úpravě je tedy možné rovnici 3.2 upravit pro výpočet afinních souřadnic v souřadném systému kamery:

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}} \\ \frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}} \end{bmatrix} \quad (3.4)$$

K nalezení matice  $\mathbf{H}$  je třeba rovnici 3.2 upravit jako  $\tilde{\mathbf{q}} \times \tilde{\mathbf{p}}\mathbf{H} = 0$ , a dále na  $\mathbf{A}_i\mathbf{h} = 0$ , kde  $\mathbf{h} = [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]$  a pro 2D DLT:

$$\mathbf{A}_i = \begin{bmatrix} -x_i & -y_i & -1 & 0 & 0 & 0 & u_i x_i & u_i y_i & u_i \\ 0 & 0 & 0 & -x_i & -y_i & -1 & v_i x_i & v_i y_i & v_i \end{bmatrix} \quad (3.5)$$

Nalezení matice  $\mathbf{H}$  je tedy možné formulovat jako optimalizační problém, kdy je třeba najít co nejlepší hodnoty parametrů  $\mathbf{h}$ :

$$\arg \min([\mathbf{A}_1^T, \dots, \mathbf{A}_N^T]^T \mathbf{h}) \quad (3.6)$$

Aby se tento postup řešení dosahoval dobrých výsledků, je třeba aby jej předcházela následující normalizace, kdy  $\mathbf{p}$  je transformován na  $\mathbf{p}'$ , tak, že průměr hodnot  $\mathbf{p}'$  je roven nule a průměrná vzdálenost hodnot od střední hodnoty  $\mathbf{p}'$  je rovna  $\sqrt{2}$ . Po provedení normalizace matice  $\mathbf{p}$ , je stejná normalizace uskutečněna i pro  $\mathbf{q}$

### Algoritmus 3D DLT

Pro 3D variantu algoritmu DLT je postup získání matice  $\mathbf{H}$  téměř identický. Dojde k navýšení počtu parametrů  $h$  z 9 na 12 a dále:

$$\mathbf{A}_i = \begin{bmatrix} -x_i & -y_i & -z_i & -1 & 0 & 0 & 0 & 0 & u_i x_i & u_i y_i & u_i z_i & u_i \\ 0 & 0 & 0 & 0 & -x_i & -y_i & -z_i & -1 & v_i x_i & v_i y_i & v_i z_i & v_i \end{bmatrix}. \quad (3.7)$$

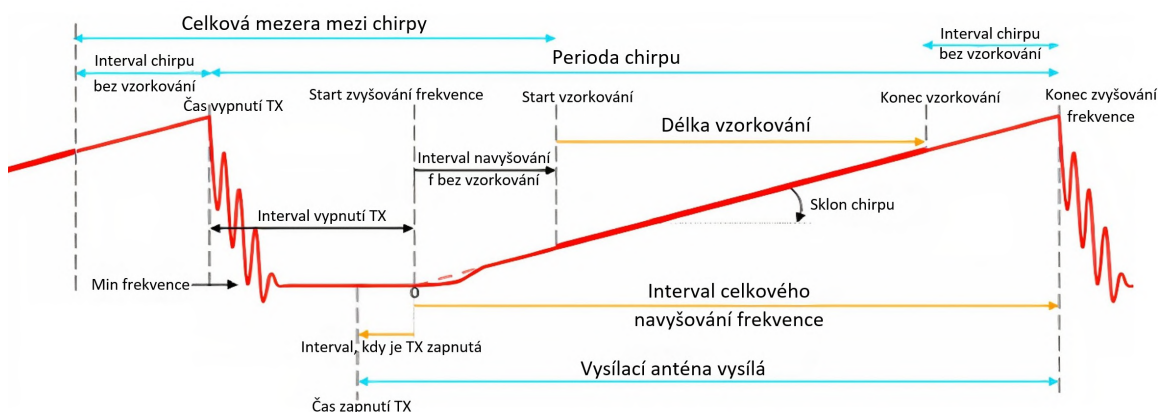
I 3D DLT musí být předcházena normalizací popsanou výše v rámci 2D DLT kalibrační metody. 3D DLT kalibraci lze použít buď při použití 2D radaru (bez informace o elevaci) a manuálního měření výšky referenčních objektů, nebo použitím 3D milimetrového radaru (poskytuje informace o elevaci).

## Kapitola 4

# Současné postupy zpracování a snímání FMCW radarem

Tato kapitola se věnuje pro tuto práci důležitým principům fungování milimetrového radaru se zaměřením na FMCW (Frequency Modulated Continuous Wave) radary a detailně popisuje postupy zpracování signálů získaných z milimetrového radaru pro účely této práce. Nejedná se tedy o přehled všech dostupných postupů, jsou zde probrány jen ty, které lze použít v této práci, kvůli maximálnímu celkovému rozsahu této práce.

Základní princip fungování milimetrového radaru je podobný radaru používanému již od druhé světové války pro vojenské účely, v dopravě a pro další případy užití. Každý radar totiž vysílá elektromagnetické pulzy, které se však mohou v mnohých parametrech lišit [5]. Základním rozdělením radarů je délka vysílaných pulzů, radary mohou vysílat buď nepřetržité elektromagnetické vlnění (CW), nebo krátké pulzy. Toto vlnění může být dále frekvenčně modulováno, nebo o stálé frekvenci [5]. Tyto popsané parametry pak určují jaké informace je možné z radarového signálu získat. V případě radarů, které vysílají nepřetržité vlnění o konstantní frekvenci (CW), je možné určit pouze rychlost na základě Dopplerova efektu, který se projeví jako změna fáze signálu. Radary vysílající krátké pulzy o konstantní frekvenci naopak dokáží určit vzdálenost objektů, nedokáží ale stanovit rychlost [5].



Obrázek 4.1: Signál vysílaný FMCW radarem<sup>1</sup>.

<sup>1</sup>Obrázek byl převzat z dokumentace [2] a upraven.



## 4.1 Stanovení vzdálenosti FMCW radarem

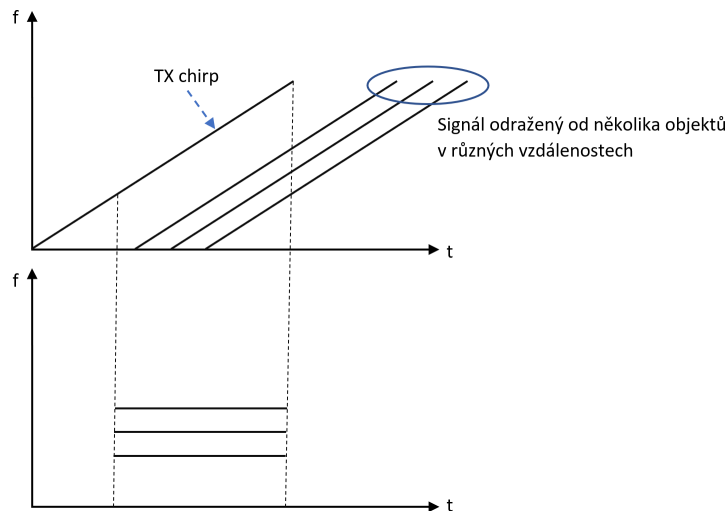
Pro stanovení rychlosti i vzdálenosti je třeba využít frekvenční modulace nepřetržitého signálu (FMCW). Signál vysílaný radarem je červeně vyobrazen na obrázku 4.1 a je možné pozorovat periodický nárůst vysílané frekvence - frekvenční modulaci, stejně jako další parametry vysílání. Jedna perioda nárůstu vysílané frekvence začínající frekvencí  $f_{min}$  a končící frekvencí  $f_{max}$  se nazývá chirp [2]. Tento signál je po odrazu od objektů přijat anténou, nebo anténami radaru. Proč je žádoucí použít více antén je vysvětleno dále. Poté je v komponentě nazývané „mixer“ přijatý signál  $x_1 = \sin(w_1 \times t + \Phi_1)$  spojen se signálem vyslaným  $x_2 = \sin(w_2 \times t + \Phi_2)$  následovně [3]:

$$x_{out} = \sin[(w_1 - w_2) \times t + (\Phi_1 + \Phi_2)], \quad (4.1)$$

čímž vznikne „beat frequency“ ( $f_b$ ) signál, který je třeba pro určení vzdálenosti detekovaných objektů a je dán vztahem [3]:

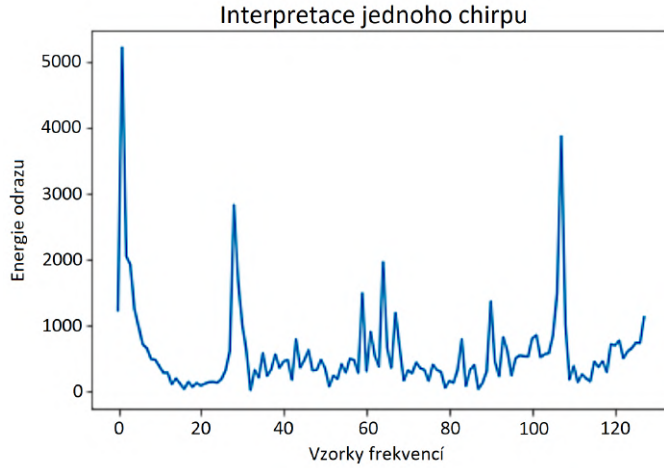
$$f_b = \frac{2ds}{c}, \quad (4.2)$$

kde  $d$  reprezentuje vzdálenost objektu odkud se signál vyslaný radarem odrazil,  $c$  je rychlost světla a  $s$  sklon růstu signálu v rámci jednoho chirpu. Jednoduchou úpravou je tedy možné na základě frekvence „beat frequency“ signálu určit vzdálenost objektů jako je ukázáno na obrázku 4.2, kde jsou identifikovatelné tři frekvence, a tedy tři objekty z „beat frequency“ signálu.



Obrázek 4.2: Ukázka „beat frequency“ signálu v časové doméně.

„beat frequency“ signál je poté možné převést z časové domény do frekvenční za pomoci Fast Fourierovy transformace (FFT) [17]. Tato transformace je nazývána jako „range FFT“, tedy vzdálenostní FFT [3]. Provedením vzdálenostní FFT je možné pozorovat jednotlivé vrcholy reprezentující detekované objekty v určité vzdálenosti vyjádřené frekvencí „beat frequency“ signálu jako na obrázku 4.3 odpovídajícímu obrázku 4.2.



Obrázek 4.3: Ukázka „beat frequency“ signálu ve frekvenční doméně pro jeden chirp.

Aby jednotlivé vrcholy po provedení vzdálenostní FFT byly rozlišitelné, je třeba aby detekované objekty byly od sebe vzdáleny o hodnotu větší, než je hodnota rozlišení vzdálenosti. Pro výpočet rozlišení vzdálenosti je možné odvodit následující vztah [17]:

$$\Delta f = \frac{2\Delta ds}{c} \quad \text{a} \quad \Delta f > 1/T_c, \quad (4.3)$$

tedy:

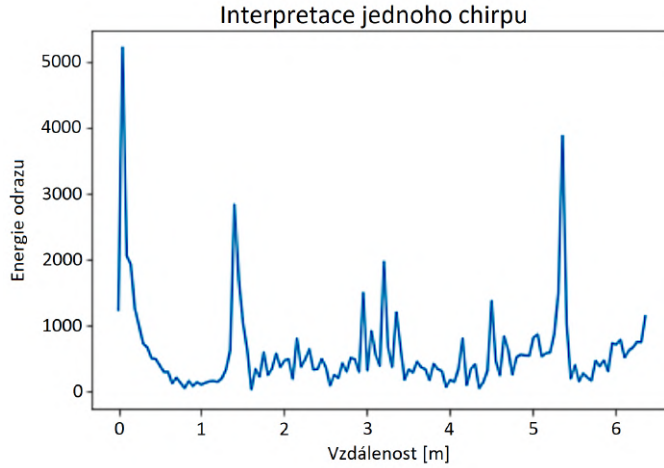
$$\frac{2\Delta ds}{c} > 1/T_c, \quad (4.4)$$

$$\Delta d > \frac{c}{2s T_c}, \quad (4.5)$$

kde  $T_c$  je délka trvání chirpu a  $2s T_c$  je šířka pásma. Délkou trvání  $T_c$  je možné zvýšit rozlišitelnost objektů ve scéně. Dále je tedy již možné určit vzdálenost objektů detekovaných radarem. Maximální detekovatelnou vzdálenost objektů je pak možné vyjádřit jako [17]:

$$d_{max} = \frac{F_s c}{2s}, \quad (4.6)$$

kde  $F_s$  je vzorkovací frekvence analogově digitálního převodníku radaru (ADC), který přijatý signál digitalizuje. Maximální vzdálenost detekce je tedy ovlivněna sklonem chirpu a vzorkovací frekvencí. Čím větší vzorkovací frekvence, tím větší maximální detekovatelná vzdálenost a čím menší sklon chirpu, tím větší maximální detekovatelná vzdálenost [2]. Na obrázku 4.4 je vyobrazen signál po vzdálenostní FFT s označením reálných vzdáleností na ose x.



Obrázek 4.4: Ukázka „beat frequency“ signálu ve frekvenční doméně převedené na reálné vzdálenosti pro jeden chirp.

## 4.2 Stanovení rychlosti FMCW radarem

Pro určení rychlosti detekovaných objektů je třeba oproti stanovení rychlosti využít více chirpů jdoucích po sobě. Pro určení vzdálenosti bylo dosud využíváno pouze jednoho chirpu, pro určení rychlosti je nutné analyzovat více chirpů jdoucích po sobě, které společně tvoří jeden snímek radaru [2]. V případě zjištění vzdálenosti byla transformace do frekvenční domény pomocí FFT prováděna na spojitém signálu, stejný princip je však aplikovatelný na diskretní signál tvořený jednotlivými chirpy pomocí Diskretní Fourierovy transformace (DFT).

Na obrázku 4.5 jsou vyobrazeny dva po sobě jdoucí chirpy v časové a frekvenční doméně, jejichž fáze se mění v závislosti na rychlosti objektů, kterým odpovídají. V rámci frekvenční domény je zřejmé, že se vrcholy nemění, vrcholy jsou totiž výsledkem vzdálenostní FFT a reprezentují vzdálenost detekovaných objektů. Mění se však jejich fáze, kterou lze dále využít pro získání rychlosti za využití principu Dopplerova efektu [17]. Rozdíl fází chirpů odpovídá vzdálenost uraženou objekty za časový úsek  $T_c$ , tedy  $vT_c$ . Změnu fáze mezi dvěma chirpy, a tím rychlost detekovaných objektů lze vyjádřit jako [17]:

$$\omega = \frac{4\pi v T_c}{\lambda}, \text{ a tedy po úpravě: } v = \frac{\lambda \omega}{4\pi T_c}, \quad (4.7)$$

kde  $\omega$  vyjadřuje změnu fáze mezi chirpy. Při použití minimálně dvou chirpů je tedy možné zjistit rychlost detekovaných objektů. Přesnost s jakou je možné rychlost určit je stanovena následovně [17]:

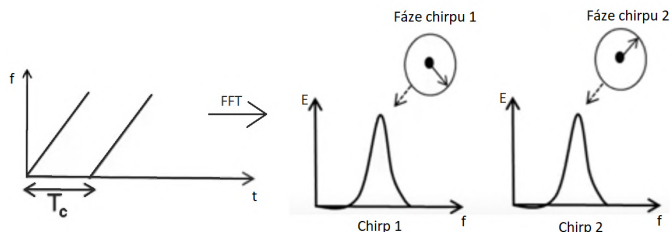
$$\Delta\omega = \frac{4\pi \Delta v T_c}{\lambda} \text{ a } \Delta\omega > \frac{2\pi}{N}, \quad (4.8)$$

tedy:

$$\Delta v > \frac{\lambda}{2NT_c}, \quad (4.9)$$

kdy  $N$  reprezentuje počet chirpů ve snímku,  $T_c$  je časový úsek mezi dvěma po sobě jdoucími chirpy. Tento výraz lze tedy dále zjednodušit, jelikož  $NT_c$  vyjadřuje celkovou délku snímku radaru ( $T_f = NT_c$ ). Přesnost s jakou je určitelná rychlost tedy nezávisí na hodnotě

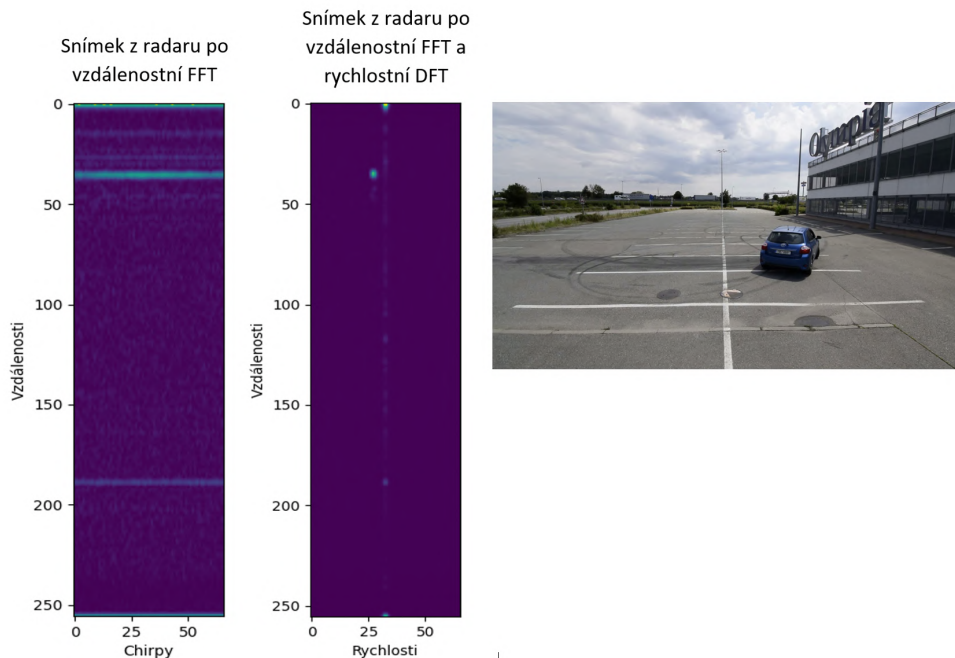
$T_c$ , záleží pouze na celkovém časovém intervalu trvání snímku radaru. Hodnota  $T_c$  však ovlivňuje maximální detekovatelnou rychlost objektů, je totiž dána vztahem:  $v_{max} = \frac{\lambda}{4T_c}$  a to kvůli jednoznačnosti určení rozdílu fází chirpů.



Obrázek 4.5: Ukázka změny fáze signálu přijatého radarem<sup>2</sup>.

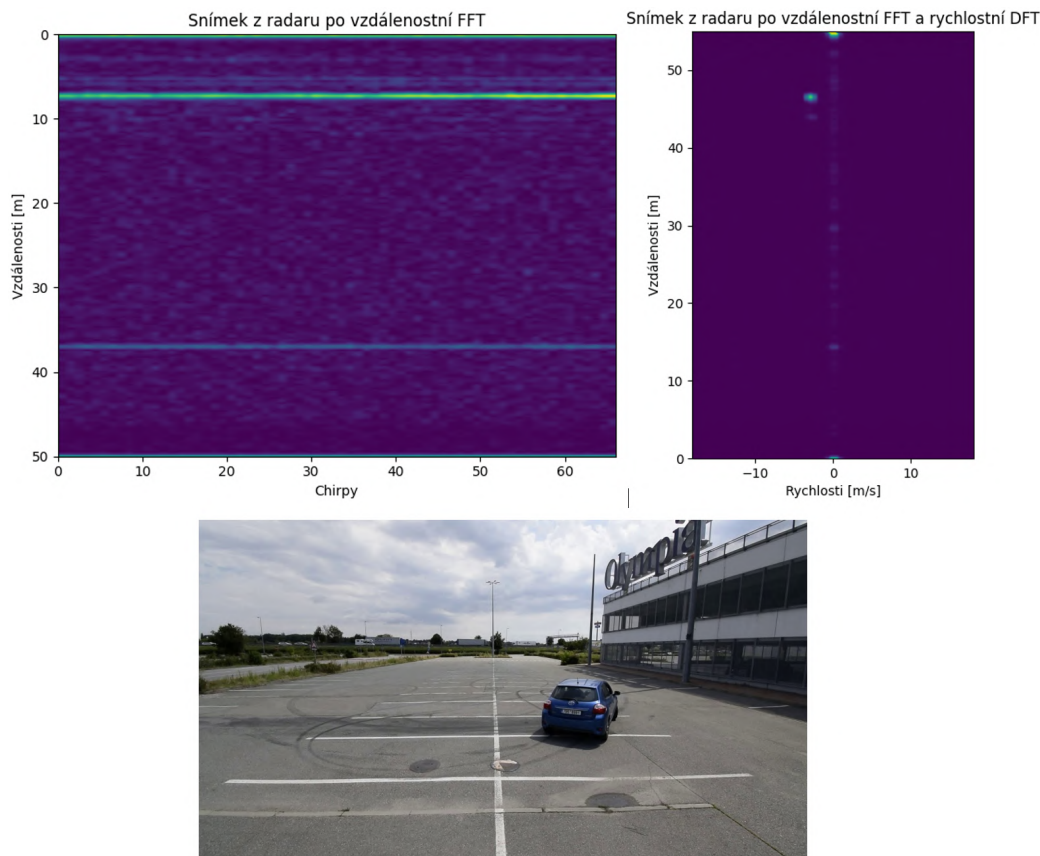
Příkladem zpracování všech chirpů tvořících radarový snímek je následující obrázek 4.6. První obrázek zleva vyobrazuje zpracování všech chirpů z jednoho snímku pomocí FFT, zde je vidět, že radar detekuje objekt v určité vzdálenosti, není ale stanovitelná jeho rychlost. Po rychlostní DFT je na druhém snímku zleva možné určit i rychlost, vrchol určuje, že se automobil na odpovídajícím snímku vzdaluje od radaru.

Z obrázku 4.6 však není možné určit v jaké vzdálenosti (v metrech) se automobil nachází, či jakou rychlostí (v metrech za sekundu) se pohybuje. Jak získat reálnou vzdálenost i rychlost s jakou se objekt k radaru přibližuje, nebo s jakou se objekt od radaru vzdaluje je již stanoveno výše a ukázáno na obrázku 4.7.



Obrázek 4.6: Ukázka změny fáze signálu přijatého radarem.

<sup>2</sup>Obrázek byl převzat z manuálu [3].



Obrázek 4.7: Ukázka změny fáze signálu přijatého radarem.

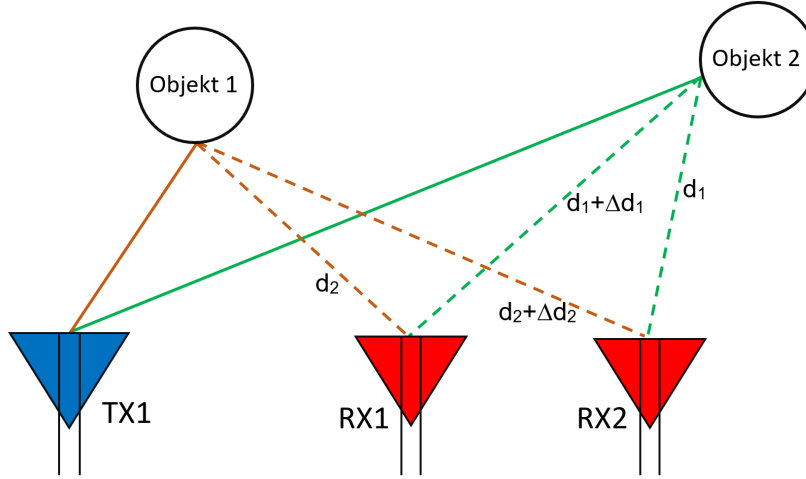
### 4.3 Stanovení směru FMCW radarem

Zjištění vzdálenosti a rychlosti objektů je pro účely počítačového vidění užitečné, chybí však určení směru, odkud odražený signál přichází, a tím společně s informací o vzdálenosti určení přesného umístění detekovaných objektů ve scéně. Pro určení směru, odkud se odražený signál z radaru vrací, je třeba využít dvě a více přijímacích antén [36]. Příklad takového rozmístění je vyobrazen na obrázku 4.8, kde je vysílací anténa označena jako TX1 a přijímací antény jako RX1 a RX2. Tyto antény jsou umístěny vedle sebe pro určení horizontálního směru a nad sebou v případě určování směru vertikálního (elevation).

Pro zjištění směru odrazu je pak využita vzdálenost, kterou musí odražený signál urazit. Tato vzdálenost se totiž bude lišit pro každou z přijímacích antén RX1 a RX2 (pokud se objekt nenachází přímo mezi přijímacími anténami) [36]. Na obrázku 4.8 je ukázána detekce dvou objektů, je tedy možné pozorovat signál vysílaný z TX1 (plné čáry) a signály odražené od objektů vracející se k radaru (přerušované čáry). Dále jsou na obrázku 4.8 označeny vzdálenosti putování odražených signálů. Pokud se tedy zaměříme ku příkladu na objekt „Objekt 2“, vidíme, že vzdálenost, kterou musel odražený signál urazit k anténě RX2 je  $d_1$  a vzdálenost k anténě RX1 je  $d_1 + \Delta d_1$ . Tato vzdálenost způsobí změnu fáze signálu, kterou je pak možné využít ke stanovení směru, odkud se odražené signály vrací [36]. Tato změna je dána následujícím vztahem:

$$\omega = \frac{2\pi\Delta d}{\lambda}, \quad (4.10)$$

kde  $\Delta d$  je již zmíněný rozdíl uražené vzdálenosti k přijímacím anténám. Tento vztah je téměř totožný se vztahem pro určení rychlosti objektů, liší se pouze koeficientem  $4\pi$  v případě rychlosti a  $2\pi$  v případě stanovení směru 4.7, jelikož se v případě rychlosti jedná o vzdálenost, kterou signál musí urazit k objektu i zpět a v případě směru je to pouze vzdálenost putování odraženého signálu.



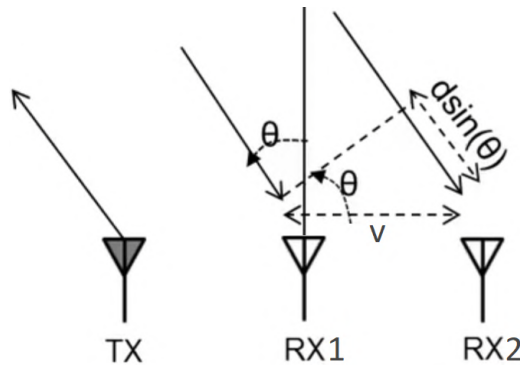
Obrázek 4.8: Ukázka změny fáze signálu přijatého radarem.

Pro určení směru je však třeba výraz 4.10 upravit tak, aby vyjadřoval směr objektů vůči radaru. Za předpokladu, že objekt je vzdálen dostatečně daleko, je možné odražené signály považovat za rovnoběžné [36]. Tím je možné upravit výraz na základě obrázku 4.9 následovně:

$$\omega = \frac{2\pi\Delta d}{\lambda} = \frac{2\pi v \sin(\theta)}{\lambda} \quad (4.11)$$

a tedy:

$$\theta = \sin^{-1} \left( \frac{\lambda\omega}{2\pi v} \right) \quad (4.12)$$



Obrázek 4.9: Ilustrace určení směru signálu pomocí dvou RX antén a jedné TX antény<sup>3</sup>.

<sup>3</sup>Obrázek byl převzat z práce [36].

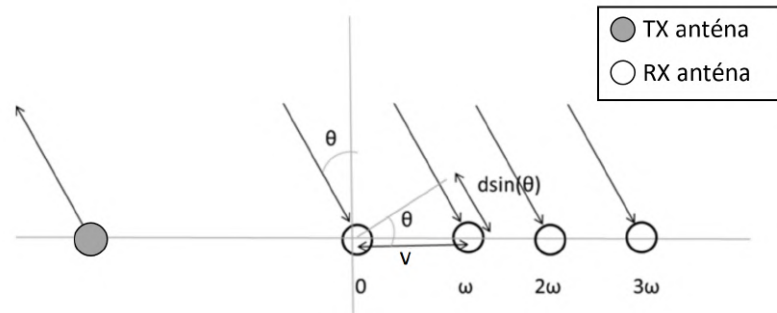
Přesnost s jakou je možné určit směr odraženého signálu, tedy rozlišit objekty nacházející se vedle sebe ve stejné vzdálenosti pohybující se stejnou rychlostí je možné odvodit následovně [36]:

$$\Delta\omega = \frac{2\pi v}{\lambda}(\sin(\theta + \Delta\theta) - \sin(\theta)) \approx \frac{2\pi v}{\lambda}\cos(\theta)\Delta\theta, \quad (4.13)$$

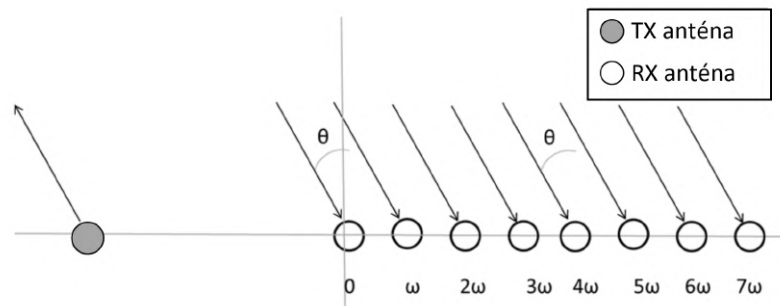
$$\Delta\omega > \frac{2\pi}{N} \text{ a tedy} \quad (4.14)$$

$$\theta_{res} > \frac{\lambda}{Nvcos(\theta)}, \quad (4.15)$$

kde  $v$  reprezentuje vzdálenost mezi přijímacími kamerami a  $N$  jejich počet. V mnoha případech jsou přijímací antény rozmístěny ve vzdálenosti  $\lambda/2$ , což po dosazení odpovídá směrové rozlišitelnosti:  $\theta_{res} = \frac{2}{N\cos(\theta)}$ , dále za předpokladu  $\theta = 0$  je výraz možné dále upravit na  $\theta_{res} = 2/N$ . Pro případ dvou přijímacích antén je pak směrovou rozlišitelnost možné vypočítat jako:  $\theta_{res} = 2/N = 2/2rad = 1rad \approx 57,3$  stupňů. Na štěstí je možné rozlišitelnost vylepšit přidáním dalších přijímacích antén, což při použití čtyř přijímacích antén jako na obrázku 4.10 vyústí v rozlišitelnost asi 28,6 stupňů a při použití osmi antén rozmístěných jako na obrázku 4.11 pak asi 14,3 stupňů. Osm a více přijímacích antén je již však mnoho, protože každá z antén vyžaduje svůj vlastní hardware, takže zvyšování směrové rozlišitelnosti tímto způsobem není ideální.



Obrázek 4.10: Ukázka možného rozmístění antén při použití 4 \* RX a 1 \* TX<sup>4</sup>.

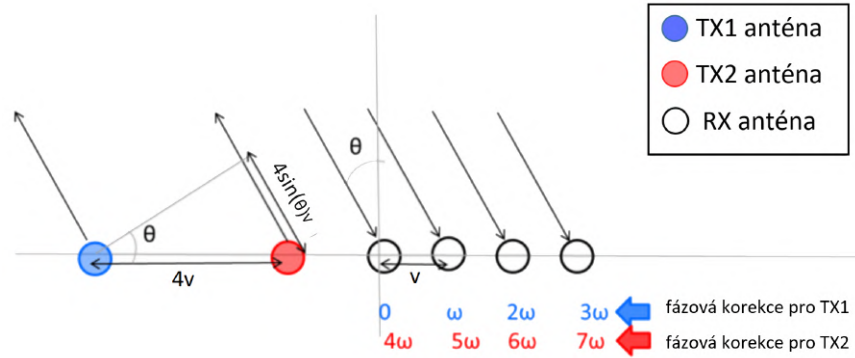


Obrázek 4.11: Ukázka možného rozmístění antén při použití 8 \* RX a 1 \* TX<sup>5</sup>.

<sup>4</sup>Obrázek byl převzat z práce [36].

<sup>5</sup>Obrázek byl převzat z práce [36].

Místo přidání přijímacích antén, je možné přidat antény vysílací [36]. Přidání jedné vysílací antény pak vede ke stejnému zlepšení rozlišitelnosti jako zdvojnásobení počtu přijímacích antén. Rozvržení na obrázku 4.12 vyústí ve stejnou rozlišitelnost jako v případě použití osmi přijímacích antén a to kvůli přidané vzdálenosti mezi vysílacími anténami, a tedy vysílací anténou TX1 a anténami RX [36]. Signál vyslaný z antény TX1 tedy urazí vzdálenost o  $4v \sin(\theta)$  větší, což vyústí v přidaný fázový posun při příjmu anténami RX. Princip přidávání vysílacích antén místo přijímacích se nazývá MIMO (Multiple-input-multiple-output) [36].



Obrázek 4.12: Ukázka rozmístění antén při použití principu MIMO ( $2 * RX$  a  $4 * TX$ )<sup>6</sup>.

Problém, který je při použití MIMO radaru třeba vyřešit, je rozdělení signálů vyslaných z TX antén. Pokud by totiž obě vysílací antény vysílaly současně, mísily by se při příjmu RX anténami. Jedním možným způsobem řešení je časově oddělit vysílání, nejprve vysílá TX1, poté TX2, tento způsob se nazývá TDM-MIMO (Time-Division-Multiplexing) [36]. Použitím TDM-MIMO se však snižuje maximální detekovatelná vzdálenost a detekční schopnost radaru. Dále pak tento postup způsobuje chybu při zjišťování rychlosti objektů [36]. Určitý počet chirpů (který odpovídá počtu použitých vysílacích antén) je vždy považován za jeden chirp s vyšší úhlovou rozlišitelností. To však ale znamená, že pro zjištění rychlosti, je třeba kompenzovat časový posun vysílání druhého a dalších chirpů, které jsou díky časovému posunu fázově a frekvenčně posunuty [36]. Tyto posuny je možné definovat následovně:

$$\Delta f = 2s dT_c/c \quad (4.16)$$

$$\Delta \phi = 4\pi dT_c/\lambda, \quad (4.17)$$

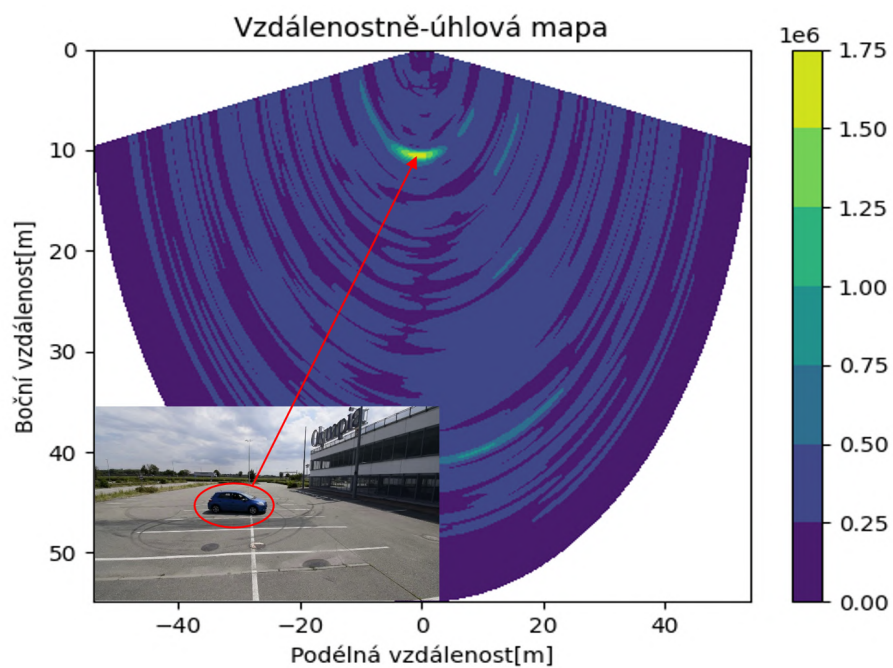
kde  $d$  odpovídá dopplerovské rychlosti a  $T_c$  je perioda vysílání po sobě jdoucích chirpů.

Dalším možným řešením použitelným pro oddělení vysílaných signálů v režimu MIMO je BPM-MIMO [36]. Tento režim umožňuje vysílání ze všech RX antén současně, vyžaduje však úpravu vysílaného signálu. BPM před vysíláním moduluje signály upravením jejich fáze buď o 0 stupňů, nebo o 180 stupňů, což při příjmu RX anténami umožňuje jejich oddělení [36].

Příkladem využití výše popsaných principů za použití čtyř RX antén a dvou TX antén v režimu TDM-MIMO je obrázek 4.13.

<sup>6</sup>Obrázek byl převzat z práce [36].





Obrázek 4.13: Ukázka vzdálenostně-úhlové mapy s odpovídajícím snímkem z kamery.

## Kapitola 5

# Zhodnocení aktuálního stavu a návrh řešení

V této kapitole jsou zhodnoceny metody, které jsou v tuto chvíli pro fúzi dat milimetrového radaru a kamery využívány. Dále jsou zhodnoceny existující datové sady a synchronizační postupy. Následuje návrh systému s architekturou konvoluční neuronové sítě určený k fúzi dat z obou senzorů. Závěrem této kapitoly je prezentováno technické zadání pro následující kapitolu 6.

### 5.1 Shrnutí dosavadního stavu metod a datových sad pro fúzi senzorů

Většina popsaných metod v části 3 zabývajících se využitím dat z obou senzorů při nasazení systému, jsou zaměřeny především na pozdní fúzi, nebo průběžnou fúzi. Neexistuje mnoho metod zaměřujících se na brzkou fúzi dat z milimetrového radaru a kamery. Většina fúzních metod pro použité senzory se řadí do kategorie průběžné fúze. Brzká fúze je však u metod zaměřujících se na použití dat pro jiné senzory využívána a měla by být použitelná i pro milimetrový radar a kameru. Metody brzké fúze využívající neuronové sítě by měly být schopné benefitovat z jednotlivých senzorů v různých situacích automaticky. Mělo by tedy být za použití brzké fúze a hlubokého učení možné navrhnout a implementovat systém, který je spolehlivější, a to zejména pro případy, kdy dochází k selhání kamery. V tu chvíli by se brzký fúzní systém měl spoléhat na data z milimetrového radaru, který není ovlivňován počasím, zhoršenými světelnými podmínkami, či poletujícími částicemi. Nemělo by při tom být nutné určovat vztahy mezi senzory apriori. Jelikož jsou však metody počítačového vidění založené na obrazových datech z kamery považovány za state of the art, měly by tyto fúzní systémy v dobrých podmínkách využít bohaté příznaky, které jsou s pomocí kamery získatelné. Milimetrový radar může být dále nápomocný i v dobrých podmínkách pro detekci objektů, které jsou například v zákrytu.

Datové sady obsahující časově synchronizovaná data z milimetrového radaru a kamery jsou popsány v části 6.4. Nejpoužívanější datová sada používaná pro účely fúze dat z milimetrového radaru a kamery je datová sada Nusenes [9]. Ta však zahrnuje pouze jednu reprezentaci dat milimetrového radaru – mračna bodů. Pokud je tedy třeba využívat jinou reprezentaci radarových dat, je třeba si vytvořit svoji vlastní datovou sadu. Jediným příkladem datové sady, která sbírá nezpracovaná data je nově vzniklá datová sada [28]. Ta je však svou velikostí stále pro účely hlubokého učení limitující.

## 5.2 Návrh metody rozpoznávání využívající fúze sensorů

Na základě zhodnocení dosavadního stavu jsem se rozhodl detekční systém spadající do kategorie brzké fúze navrhnout, otestovat a následně porovnat s metodou spoléhající se pouze na obrazová data z kamery. Pro tento účel jsem musel vytvořit datovou sadu pro účely trénování a testování. K vytvoření datové sady pak bylo třeba určit způsob získání dat z kamery, milimetrového radaru a následný proces zpracování dat z radaru. Musel jsem data ze sensorů časově synchronizovat, kalibrovat a správně reprezentovat. Zpracování a reprezentace dat je důležitá především pro milimetrový radar, kdy je k dispozici nepřehledné množství možných reprezentací dat.

## 5.3 Vhodné případy pro ověření navržené metody

Mezi vhodné případy patří především využití sensorů v dopravě pro účely asistentů pro řízení vozidel. Stejně tak je ale možné milimetrový radar a kameru použít pro monitorování dopravy. Dalšími možnými případy užití mohou být, jak je popsáno v části 6.4, rozpoznávání pohybů ruky, sledování pozice hlavy řidiče.

V této práci jsem se rozhodl využít milimetrový radar a kameru pro potřeby monitorování dopravy pro spolehlivější detekci objektů ve scéně ve zhoršených podmínkách. Následuje technické zadání řešeného problému, tedy přesné stanovení úkolů potřebných k vytvoření fúzního detekčního systému pro potřeby dopravy.

## 5.4 Technické zadání

Na základě výše uvedené analýzy a možností řešení jsem se rozhodl vytvořit systém, který bude splňovat:

- Systém bude využívat data z milimetrového radaru a monokulární kamery.
- Pro účely detekce bude tento systém využívat konvoluční neuronovou síť.
- Zaměřením systému bude detekce objektů ve scéně.
- Systém bude využívat brzkou fúzi dat, tedy bude data fúzovat před vstupem do detekční konvoluční neuronové sítě.
- Trénování a testování systému bude realizováno na vlastní datové sadě obsahující časově synchronizovaná a kalibrovaná data z různých dopravních situací z obou sensorů.
- Vytvořený systém bude robustnější pro data snímaná ve zhoršených podmínkách, než systém využívající pouze obrazová data.
- Systém bude na vstupu přijímat data z milimetrového radaru reprezentované mračnem bodů, nebo vzdálenostně-úhlovou mapou.
- Pro vyhodnocení úspěšnosti detekce budou využity metriky běžně používané pro ohodnocení jiných detekčních systémů.
- Tvorba fúzního systému bude realizována pomocí programovacího jazyka Python a vhodných knihoven, modulů a frameworků. Nejdůležitějšími specifikovanými prostředky jsou framework PyTorch a knihovna OpenCV.

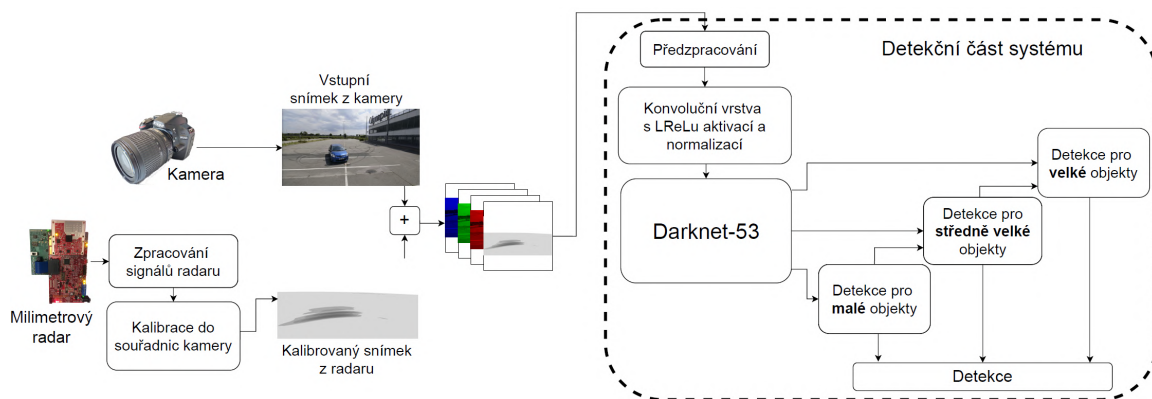
## Kapitola 6

# Implementace a testování

Tato část se věnuje popisu navrženého fúzního systému, jeho testování a úkolům s tím souvisejících. Mezi úkoly, které bylo třeba řešit, a které jsou v této části popsány, patří časová synchronizace a kalibrace senzorů, proces získávání dat z použitých senzorů, zpracování dat milimetrového radaru, reprezentace dat milimetrového radaru a vytvoření datové sady.

Výše zmíněné úkoly jsou nutným předpokladem pro trénování a testování výsledného fúzního detekčního systému. Implementovaný fúzní systém je závěrem této kapitoly testován oproti detekční metodě YOLOv3 využívající pouze vizuální data z kamery. Oba srovnávané systémy jsou pak trénovány a vyhodnocovány na totožné datové sadě.

Na následujícím obrázku 6.1 je vyobrazeno blokové schéma navrženého fúzního systému. Vstupem detekční části jsou časově synchronizované a kalibrované snímky z milimetrového radaru a kamery, které jsou pro účely vstupu do detekční části konkatenovány. Tyto snímky jsou vytvořeny pomocí zpracování signálů radaru a následné kalibrace. Časovou synchronizaci je nutno zajistit již při snímání, proto není na schématu vyobrazena. Výstupem detekční části systému jsou ohraničující rámečky detekovaných objektů s přiřazenými třídami pro dané snímky ze senzorů.



Obrázek 6.1: Diagram navrženého fúzního systému.

Dle schématu 6.1 jsem tedy implementoval detekční část systému pomocí programovacího jazyka Python a frameworku PyTorch. Zpracování signálu radaru jsem provedl pomocí MatLab skriptu, který hodnoty vysílané radarem rozdělí na jednotlivé snímky a provede vzdálenostní FFT, a dále Python skriptem, který provede rychlostní FFT a určení směru společně s fázovou kompenzací kvůli použití TDM MIMO (pro více informací o zpracování

radarových signálů viz kapitolu 4). Kalibrace zpracovaných radarových dat je pak provedena pomocí Direct Linear Transform (DLT).

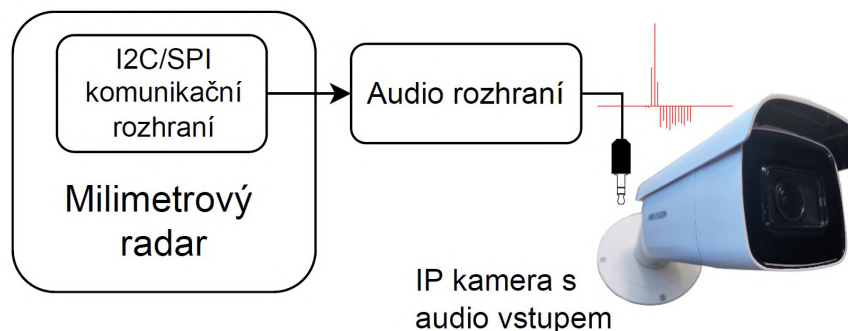
Pro trénování a testování detekčního systému jsem vytvořil vlastní datovou sadu, která se skládá z nezpracovaných signálů radaru časově synchronizovaných se snímky z kamery. Pro účely ověření detekční schopnosti fúzního systému jsem dále simuloval pro obrazová data zhoršené podmínky ve scéně. Pro vytvořený systém jsem poté provedl vyhodnocení pomocí vhodných metrik a srovnal jeho schopnosti s detekčním systémem využívajícím pouze obrazová data. Popsané kroky jsou dále rozebrány v následujících částech této kapitoly.

## 6.1 Popis synchronizace dat z radaru a kamery

V části 3.4 jsou čtenáři k nahlédnutí některé z dostupných metod pro časovou synchronizaci, ve zmíněné části se také nachází popis využití nejrůznějších příznaků z různých senzorů pro synchronizaci. Patří mezi ně například využití vibrací jedoucího automobilu, či trajektorie pohybu objektů v obraze. V této části navrhuji časově synchronizační postup využívající audio kanál kamery a komunikační rozhraní milimetrového radaru pro přesnou synchronizaci snímků těchto senzorů. Navazuji zde na svou předchozí práci Exploitation of Neural Networks for Fusion of Camera and Millimeter-Wave Radar Data [41], kde jsem tento synchronizační postup poprvé navrhl.

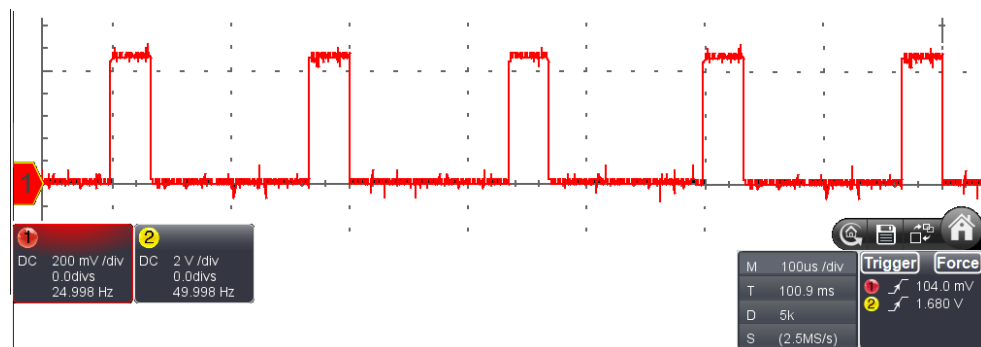
### Popis navržené synchronizační metody

Na obrázku 6.2 je vyobrazen diagram navržené synchronizační metody. Milimetrový radar je zde skrze nízkoúrovňové komunikační rozhraní a audio rozhraní připojen ke zvukovému vstupu IP kamery. Pro potřeby navržené metody se však může jednat o jakoukoliv kameru s audio vstupem. Blok „Audio rozhraní“ slouží k převodu signálů vyslaných z milimetrového radaru na audio signál, který vyhovuje parametrům vstupního audio signálu, tedy úrovně odpovídá výstupu konvenčního mikrofону.



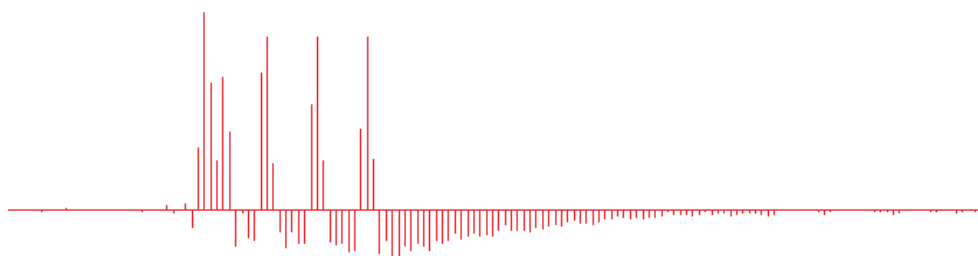
Obrázek 6.2: Diagram navržené synchronizační metody.

Principem metody je pak úprava software milimetrového radaru tak, aby pokaždé při zahájení snímání nového snímku vyslal signál skrze nízkoúrovňové rozhraní (SPI, UART, I2C). Tento signál musí být alespoň částečně interpretovatelný jako audio signál. Radar tedy může vyslat signál jako na obrázku 6.3.



Obrázek 6.3: Příklad signálu vyslaného milimetrovým radarem pro účely synchronizace.

Radarem vyslaný signál je poté upraven, aby byl zřetelný v rámci audio kanálu kamery. Příkladem signálu vyslaného milimetrovým radarem, zpracovaným audio rozhraním a přijatého kamerou je ukázán na obrázku 6.4.

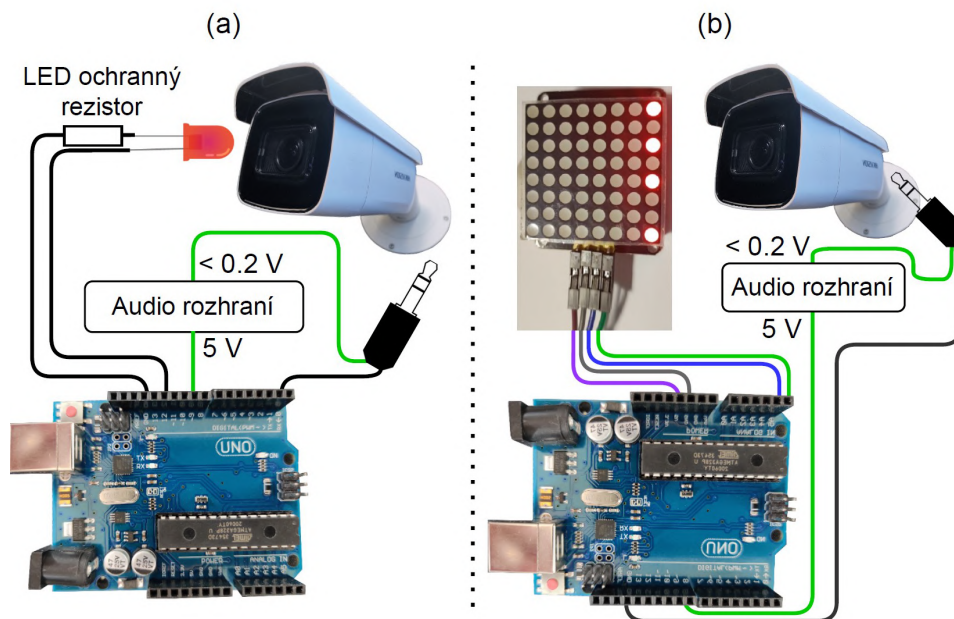


Obrázek 6.4: Příklad signálu (vyslaného radarem) zaznamenaného na audio vstupu kamery.

Takto přijatý audio signál je poté možné využít jako časovou značku označující začátek snímání radaru, a tedy ve video signálu kamery vyhledat odpovídající snímek z kamery. Popsaný způsob tedy může sloužit k automatické synchronizaci nahrávek z radaru a kamery na základě analýzy zvukového kanálu kamery. Potenciálním problémem navrženého postupu je však interní synchronizace kanálů kamery.

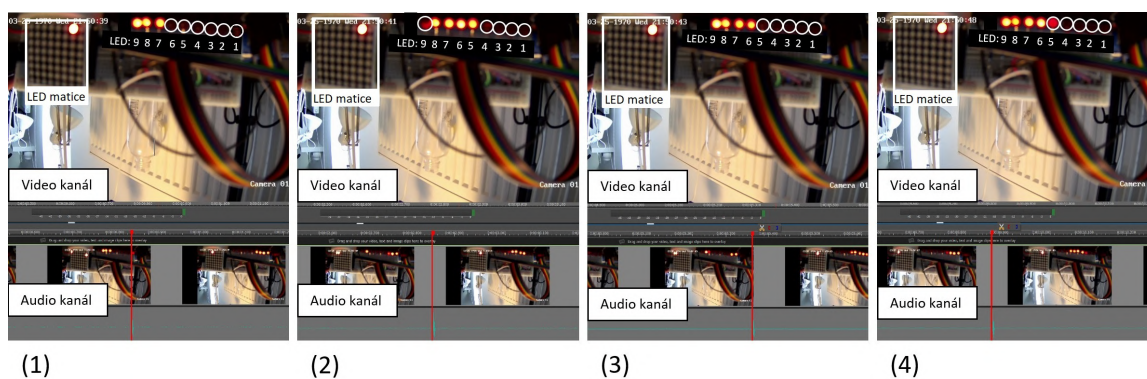
### Ověření přesnosti interní synchronizace kamery

Pro ověření přesnosti synchronizace kanálů kamery jsem navrhl obvody ukázané na obrázku 6.5. Pomocí těchto obvodů je možné určit jak jsou kanály kamery synchronizovány, jaké relativní konstantní zpoždění vykazují a u u obvodu (b) dále variabilitu zpoždění kanálů. Obvod (a) lze považovat za prvotní experiment a nebude dále při ukázce ověření přesnosti využíván. Obvod (b) je oproti obrázku 6.5 částečně upraven, společně s LED maticí jsem při experimentech využíval 10 LED diod připojených přímo k Arduino vývojové desce pro zvýšení přesnosti měření.



Obrázek 6.5: Obvody pro ověření přesnosti interní synchronizace audio a video kanálu kamery, (a) obvod s jednou LED, (b) obvod s LED maticí pracující jako čítač.

Pro určení chyby synchronizace kanálů kamery jsem využil vývojovou desku Arduino UNO, která ovládala LED diodu (obvod (a)), nebo více LED diod současně (obvod (b)) a zároveň vysílala signál na audio vstup kamery. Zobrazením diod před objektivem kamery bylo poté možné určit časové značky změny stavu diod a čas, kdy byl kamerou přijat audio signál. V případě obvodu (b) je možné určit z jednoho videozáznamu jak konstantní složku zpoždění, tak variabilitu, jelikož jsou v tomto případě diody použity jako čítač s malým časovým krokem. Je tedy z video kanálu kamery možné vyčíst aktuální hodnotu tohoto čítače a tím určit posun kanálů kamery pro každý snímek.



Obrázek 6.6: Příklady kamerového výstupu pro obvod ověření přesnosti (b) s první řadou LED diod nahrazenou přímo připojenými diodami.

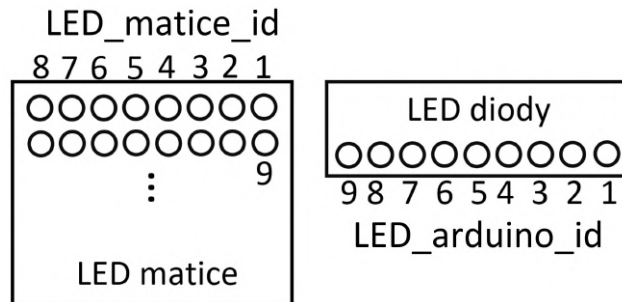
Na obrázku 6.6 jsou vyobrazeny příklady videozáznamu pořízeného pomocí obvodu (b) z obrázku 6.5 s výše popsanou úpravou, kdy je řada diod matice nahrazena diodami přímo připojenými k vývojové desce Arduino. Každá z těchto přímo připojených diod reprezentuje časový krok 5 ms a změna diody LED matice pak reprezentuje časový krok 50 ms. Díky

tomu je možné určit posun kanálů kamery s maximální přesností 5 ms v rámci audio kanálu je zřetelný přijatý signál z Arduino vývojové desky a v rámci video kanálu je zobrazen odpovídající snímek. Z LED matice a LED diod je pak možné vyčíst časovou značku, která odpovídá zaznamenání audio signálu kamerou. Tento audio signál je Arduinem vyslaný každých 500 ms.

Časová značka  $T$  je dána jako:

$$T = \text{LED\_arduino\_id} * 5ms + \text{LED\_matice\_id} * 50ms, \quad (6.1)$$

kdy na obrázku 6.7 je vyobrazeno použité rozvržení diod:



Obrázek 6.7: Schéma diod pro čtení časových značek.

Z obrázku příkladů 6.6 je tedy možné vyčíst, že snímky (1) až (4) zobrazují časové značky  $T$  za předpokladu použití nejvyšší hodnoty časové značky, kdy se na snímcích zobrazuje více aktivních LED diod kvůli expozičnímu času kamery (v případě příkladu (2) není jisté, zda by měla být přímo připojená LED dioda s  $id = 9$  považována za aktivní):

- pro snímek (1)  $T = 95ms$ ,
- pro snímek (2)  $T = 90ms$ ,
- pro snímek (3)  $T = 95ms$  a
- pro snímek (4)  $T = 95ms$ .

Dále je tedy možné určit zpoždění audio kanálu kamery s přesností 5ms. Z 20 snímků pořizovaných v čase přijetí audio signálu (18 z nich vykazuje zpoždění  $T = 95ms$ , jeden snímek  $T = 90ms$  a jeden snímek  $T = 100ms$ ) je tedy možné určit, že konstantní zpoždění ( $constT$ ) audio kanálu je  $constT = 90ms$  s průměrnou absolutní odchylkou  $varT = 0,5ms$  a maximální absolutní odchylkou  $absT = 5ms$ .

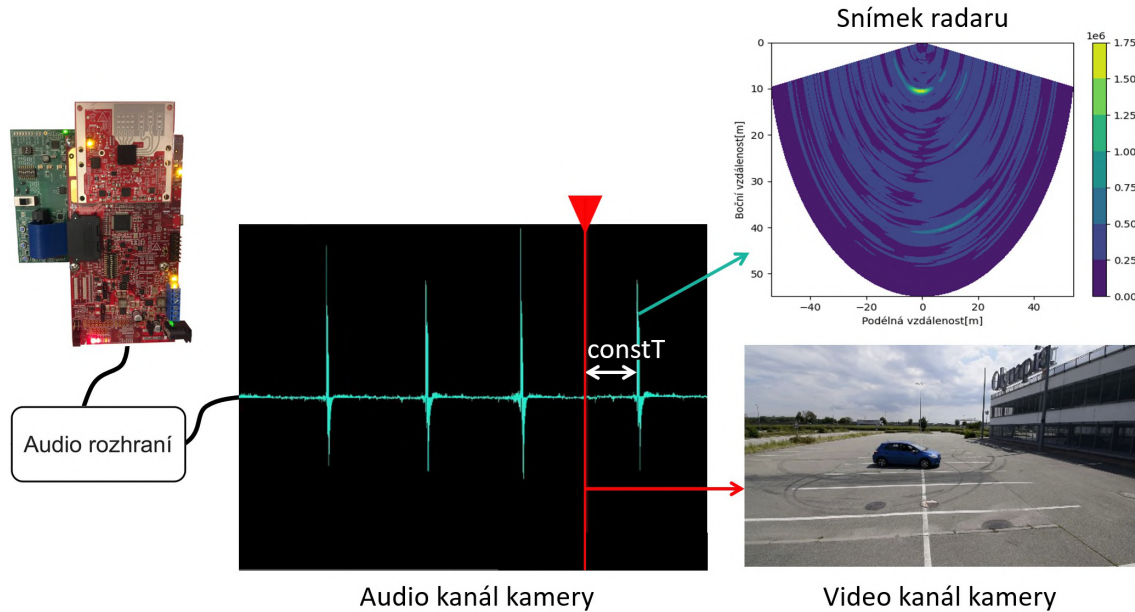
Interní synchronizace zkoumané kamery tedy sice vykazuje konstantní zpoždění audio kanálu, hodnota průměrné odchylky zpoždění je však nízká a je tedy možné zpoždění audio kanálu v rámci navržené synchronizační metody kompenzovat.

## Implementace navržené synchronizační metody

Navrženou časovou synchronizační metodu lze realizovat za pomoci analýzy audio kanálu videozáznamu, kdy detekované lokální maximum audio signál odpovídá signálu vyslanému milimetrovým radarem a zpracovanému audio rozhraním. Snímek ve video kanálu kamery je nalezen na základě časové značky lokálního maxima audio signálu, která je posunuta



o konstantní zpoždění ( $constT$ ) nalezené v předchozí části. Tím je možné vyhledat odpovídající snímky z milimetrového radaru a kamery. Tento proces je ilustrován na obrázku 6.8. Časová značka daná lokálním maximem audio signálu k amery je využita pro nalezení i-tého snímku z radaru a časová značka ilustrovaná vertikální červenou čarou odvozená od značky dané audio signálem za pomoci hodnoty  $constT$  je použita pro výběr odpovídajícího snímku kamery.



Obrázek 6.8: Ilustrace procesu výběru odpovídajících snímku z radaru a kamery pomocí navržené synchronizační metody.

## 6.2 Způsob získání referenčních výstupních hodnot

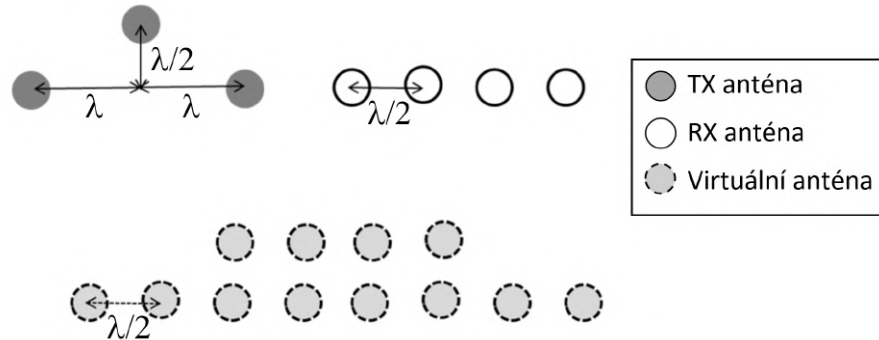
Pro potřeby získání datové sady pro trénování navržených metod jsem vytvořil vlastní datovou sadu, která obsahuje nezpracované signály sesbírané z milimetrového radaru synchronizované se snímky z kamery. Jak je v části 6.4 popsáno, z takovýchto nezpracovaných dat je možné vytvořit jakoukoliv reprezentaci dat z milimetrového radaru. Dále je v části 6.4 uvedeno, že existuje pouze jedna volně dostupná datová sada RadICaL [28], která je tvořena nezpracovanými radarovými signály. Ta však při psaní této diplomové práce teprve vznikala, je svou velikostí limitující a je zaměřena spíše na vytváření systémů pro řízení vozidla, či autonomního řízení vozidla. V této části je tedy popsán způsob a prostředky potřebné pro získání vlastní datové sady, která je pak dále rozebrána v části 6.4.

### Hardware použitý pro získání referenčních hodnot

Zde jsou rozebrány hardwarové prostředky použité pro snímání společně s rozvržením senzorů a prostředky potřebnými pro realizaci navrženého rozmístění a připojení senzorů.

## Milimetrový radar Texas Instruments IWR6843ISK

Použitým milimetrovým radarem je modul FMCW milimetrového radaru Texas Instruments IWR6843ISK Rev. D. Tento modul milimetrového radaru pracuje pro frekvence 60 GHz až 64 GHz a poskytuje tři vysílací a čtyři přijímací antény v rozvržení ukázaném na obrázku 6.9, které při použití v režimu MIMO vytvoří virtuální anténové pole dvanácti antén pro určení jak horizontálního, tak vertikálního směru odraženého signálu. V režimu MIMO se tedy jedná o horizontální směrovou rozlišitelnost danou vzorcem  $\theta_{res} = 2/N$  (jelikož  $v = \lambda/2$ ), a tedy 14.3 stupňů, více viz v části 4.



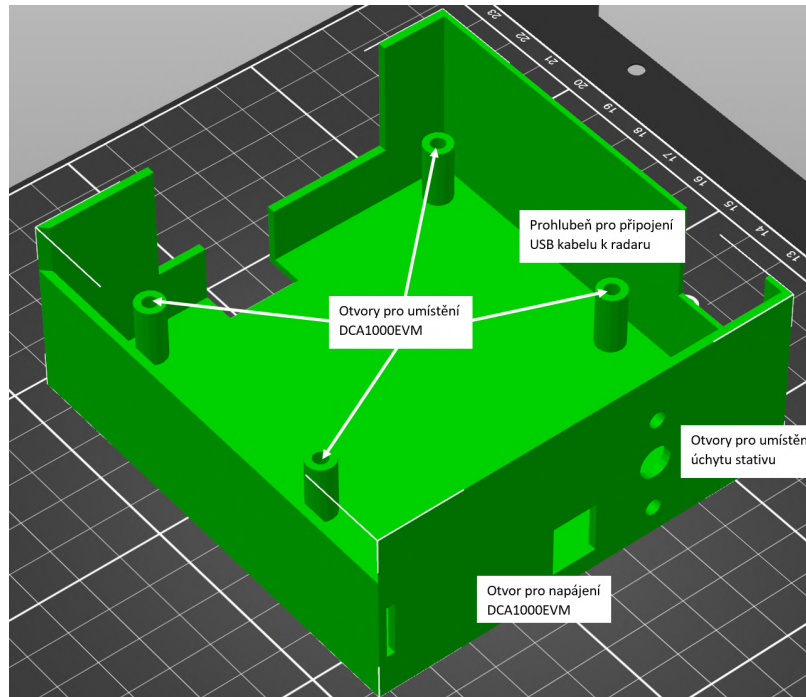
Obrázek 6.9:

## Snímací karta DCA1000EVM

Pro získání nezpracovaných signálů z milimetrového radaru je třeba z radarového modulu přenášet data do počítače přes snímací kartu pomocí Ethernet rozhraní. Modul IWR6843ISK sice disponuje USB rozhraním, nedokáže však přenášet nezpracované signály, ale pouze mračna bodů, či vzdálenostně-rychlostní, nebo vzdálenostně-úhlovou teplotní mapu pro statické objekty s malou snímací frekvencí do 5 snímků za vteřinu.

DCA1000EVM snímací karta umožňuje do počítače přenášet všechna nezpracovaná data i pro snímací frekvence větší než 20 snímků za vteřinu. Tím je možné zpracovat radarové signály libovolným způsobem a vytvořit jakoukoliv reprezentaci signálů, tedy i ty, které by nebylo možné přenášet přímo z radarového modulu přes USB rozhraní.

Pro umístění snímací karty a radarového modulu jsem pro uchycení na stativ vytvořil 3D model ukázan s popisky na obrázku 6.10 a na 3D tiskárně vytisknutý a osazený na obrázku 6.11, kde zelený plošný spoj je snímací karta (DCA1000EVM) a červený plošný spoj pak radarový modul (IWR6843ISK Rev. D).



Obrázek 6.10:



Obrázek 6.11:

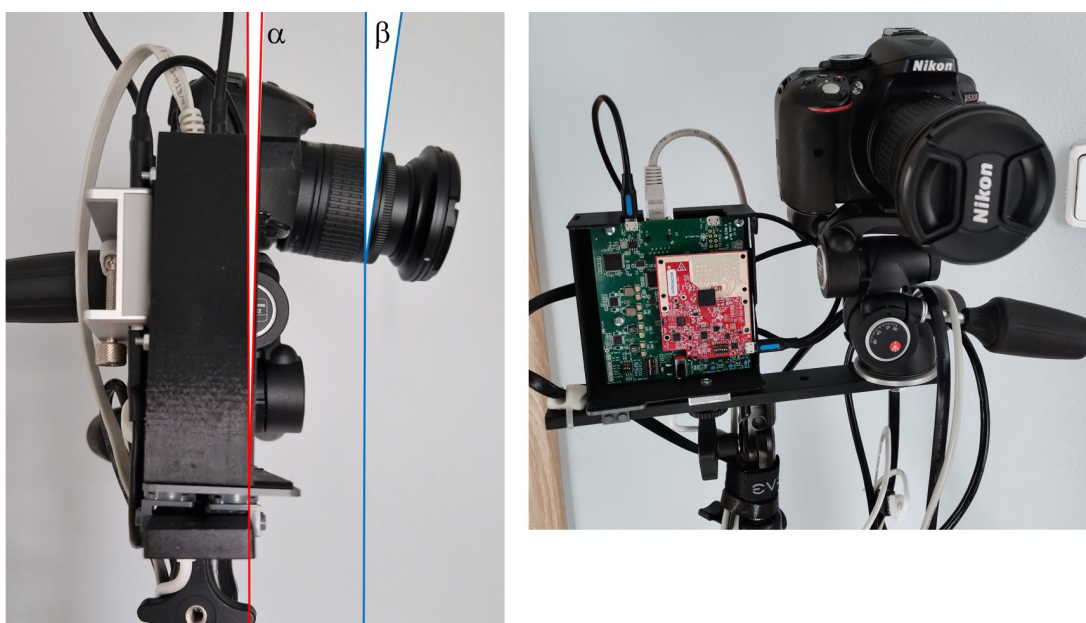
### Monokulární kamera Nikon D5300

Pro účely sběru obrazových dat jsem použil fotoaparát NIKON D5300 podporující přenos obrazu do počítače pomocí HDMI připojení a širokoúhlý objektiv s ohniskovou vzdáleností

10 mm. Za pomoci těchto komponent je možné přenášet video o rozlišení  $1920 \times 1080$  pixelů a o snímací frekvenci až 30 snímků za vteřinu.

### Rozmístění senzorů pro snímání

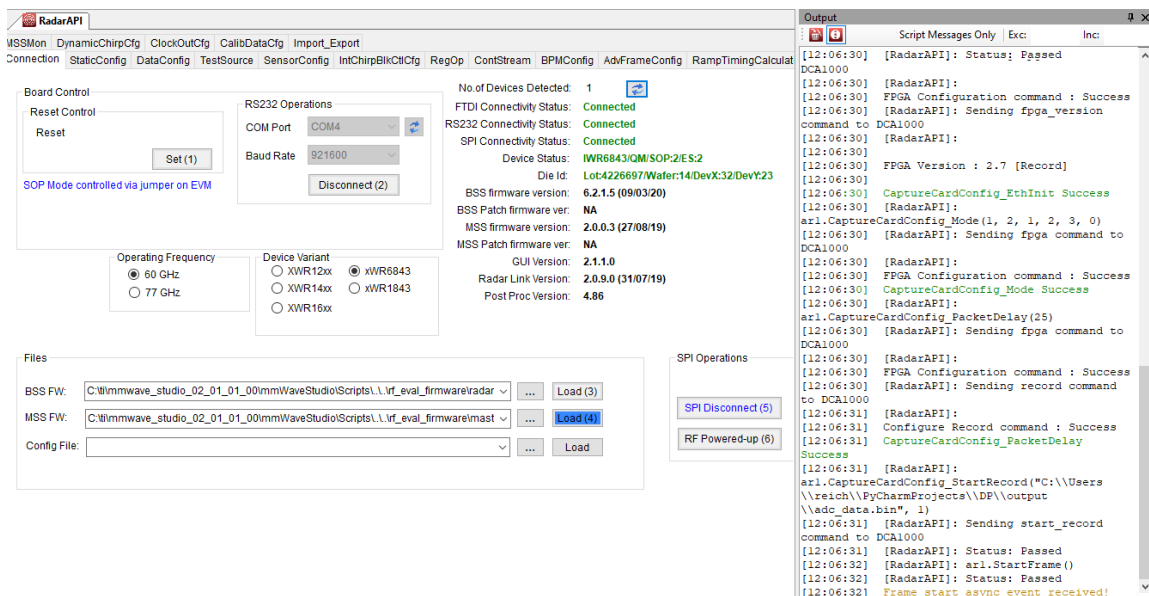
Tato část se dále zabývá vhodným rozmístěním senzorů pro snímání. Kamera a milimetrový radar jsem umístil na stativ do výšky 2,31 m pro radar (měřeno od středu přijímacích antén) a 2,39 m pro fotoaparát (měřeno od středu obrazového snímače fotoaparátu) vedle sebe 6.12. Radar při tom směřuje směrem dolů pod úhlem  $\alpha = -1$  stupňů oproti kolmici k podložce a fotoaparát směřuje dolů pod úhlem  $\beta = -4,5$  stupňů oproti kolmici k podložce (obrázek 6.12). Umístěním do výšky nad 2,3 metru je možné lépe snímat dopravní situaci a zároveň je výška vhodná pro příklady použití milimetrového radaru od Texas Instruments pro monitorování dopravy. Radar stejně jako kamera tímto způsobem dokáží detekovat objekty, které by při umístění senzorů níže, byly v zákrytu.



Obrázek 6.12:

### Software použitý pro získání referenčních hodnot

Pro snímání byl použit program MMWAVE-STUDIO (ukázáno na obrázku 6.13), který zajišťuje komunikaci milimetrového radaru a snímací karty DC1000EVM s počítačem. Pomocí FTDI USB rozhraní je možné snímání radaru a snímací karty softwarově ovládat, je možné zahájit snímání, nastavit konfiguraci přenosu dat ze snímací karty, parametry chirpů milimetrového radaru. Nezpracovaná data z milimetrového radaru jsou pak skrze snímací kartu do počítače odesílána pomocí Ethernet rozhraní. Zahájení snímání je uskutečněno za pomoci LUA skriptu, který skrze MMWAVE-STUDIO umožňuje automatické nastavení snímací karty a milimetrového radaru (přes FTDI USB rozhraní), následné zahájení snímání a sběr dat. Pomocí tohoto LUA skriptu a MMWAVE-STUDIO je tedy možné ovládat milimetrový radar a přijímat nezpracovaná data z radaru.



Obrázek 6.13: Snímek MMWAVE-STUDIO při začátku snímání milimetrovým radarem.

Pro sběr dat z k počítači připojené kamery je pak vytvořen Python skript, který kromě snímků z kamery dále zaznamenává i časové značky určené na základě času přijetí snímků počítačem. Tyto značky je pak možné využít pro synchronizaci milimetrového radaru a kamery. Při snímání tedy nebylo využito synchronizačního postupu navrženého v rámci části 6.1. Délka jednotlivých měření je totiž 30s, nebo 60s. Jedná se tedy o krátké časové úseky a není tím třeba časovou synchronizaci navrženým postupem 6.1 využít. Maximální odchylka frekvence snímání radaru a kamery nenabývá za tak krátký časový úsek velké odchylky. Navržený synchronizační postup 6.1 je vhodný pro použití při nepřetržité snímání.

Časové značky přijetí snímku z milimetrového radaru jsou určeny pomocí programu tshark<sup>1</sup>. Ukázalo se totiž, že přijímání paketů s nezpracovanými daty ze snímací kamery není možné přijímat za pomoci Python skriptu. Tento postup totiž snímací kartou vysílané UDP pakety nedokázal přijmout všechny a docházelo k jejich ztrátě.

### 6.3 Popis kalibrace radarových a obrazových dat

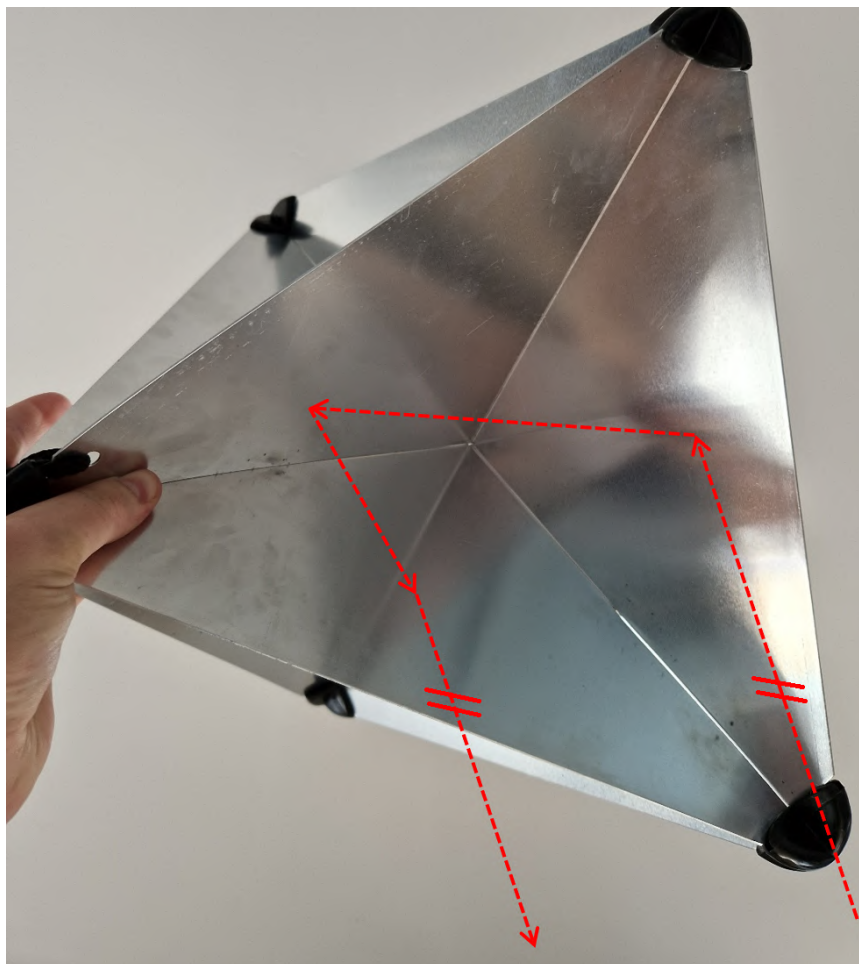
Tato část popisuje použitý postup kalibrace, který jsem využil, potřebné prostředky a vyhodnocuje přesnost měření vzdálenosti radarem společně s přesností kalibrační metody.

#### Použité kalibrační prostředky

Pro účely kalibrace senzorů jsem využil metodu Direct Linear Transform (DLT), která je popsána v části 3.5. Tato metoda dokáže transformovat body ze souřadného systému radaru přímo do systému souřadnic kamery. Bylo tedy třeba určit referenční kalibrační body pro milimetrový radar a kameru, které by bylo dále možné použít pro potřeby kalibrace. K tomuto účelu jsem použil koutový odražeč používaný např. ve studii zaměřující se na

<sup>1</sup>Program tshark je tzv. packet sniffer a umožňuje odposlech síťové komunikace na určitém rozhraní, IP adresy a portu pro daný druh paketů.

porovnání synchronizačních postupů pro milimetrový radar a kameru [34]. Koutový odražeč je totiž dobře pozorovatelný jak na snímcích z kamery, tak z milimetrového radaru a je ukázán na obrázku 6.14, kde je také vyobrazen jeho princip. Při příchodu signálu z radaru, je totiž tento signál díky rozvřzení ploch odražen rovnoběžně zpět k milimetrovému radaru, a tím je radarem dobře pozorovatelný.



Obrázek 6.14: Rohový odražeč pro získání kalibračních bodů.

### Popis získání transformační matice kalibrací

Při stacionárním umístění kamery a radaru v pozici popsané v předchozí části 6.2 jsem pořídil 10 snímků pro různé umístění odražeče ve scéně. Tyto pozice jsem přesně zaznamenal pomocí laserového dálkoměru pro ověření správnosti určení vzdálenosti radarem. Vzdálenosti pro 9 z 10 měření jsou ukázány v rámci tabulky 6.1, u jednoho z měření došlo k chybě záznamu dat milimetrovým radarem.

Z odpovídajících snímků z milimetrového radaru pak bylo možné určit pozice odražeče v souřadném systému radaru. V rámci této práce využívám pouze 2D reprezentaci dat z milimetrového radaru. Kvůli tomu není možné určit výšku, v jaké jsou odražeče umístěny nad zemí, je tedy možné stanovit pouze podélnou a boční vzdálenost objektů.

ID měření	Podélná vzdálenost [m]	Boční vzdálenost [m]	Výška [m]
1	20	0	1,25
2	12,5	5	0,55
3	10	-5	1,55
4	5	5	1,55
5	20	10,75	1,3
6	10	-5	0,9
7	5	2,5	1,6
8	15	-6,62	0,15
9	5	0	1,4

Tabulka 6.1: Tabulka s reálnými umístěními koutového odražeče ve scéně.

ID měření	Podélná vzdálenost [m]	Boční vzdálenost [m]
1	20,1	-2,1
2	12,7	3,7
3	9,9	-6,11
4	5	4,52
5	20,3	8,7
6	10,1	-6,12
7	4,9	1,97
8	15	-8,2
9	5,1	-0,51

Tabulka 6.2: Tabulka s umístěními koutového odražeče zjištěnými milimetrovým radarem.

Z tabulky 6.1 a tabulky 6.2 je možné ověřit, s jakou chybou radar dokáže stanovit vzdálenost objektů. Je tedy možné určit, že průměrná relativní odchylka ( $\epsilon_{mean}$ ) stanovení podélné vzdálenosti, odpovídá:

$$\epsilon_{mean} = \frac{\sum |\hat{x} - x|}{n} = 0,11m, \quad (6.2)$$

kdy  $n = 9$  a  $x$  odpovídá podélným rychlostem určeným radarem a  $\hat{x}$  pak reálným umístěním. Maximální chyba  $\epsilon_{max}$  pak nabývá hodnoty  $\epsilon_{max} = 0.3m$  pro měření  $ID = 5$ . Maximální chybu s jakou by měl radar pracovat ( $\epsilon_{maxP}$ ) lze určit na základě vztahů z části 4 a znalosti parametrů použitých při měření. Takto vypočtená maximální chyba by měla nabývat hodnoty  $\epsilon_{maxP} = 0,1085m$ , což experimentálnímu ověření zcela neodpovídá. Vyšší hodnota maximální chyby měření je však pravděpodobně způsobena chybou stanovení reálného umístění odražeče ve scéně, kdy pro větší vzdálenosti bylo třeba vzdálenosti určovat skládáním několika měření kvůli maximálnímu měřitelnému dosahu použitého dálkoměru.

Příklady referenčních snímků jsou vyobrazeny na obrázku 6.15. K určení transformační matice pro potřeby kalibrace bodů z milimetrového radaru do systému souřadnic kamery pak lze určit buď pomocí 2D DLT, nebo 3D DLT pokud jsou použity společně s body určenými radarem naměřené výšky umístění koutového odražeče (změřeny pomocí laserového

dálkoměru). Transformační matice určené pomocí 2D DLT ( $T_{2D}$ ) a 3D DLT ( $T_{3D}$ ) vykazují pro devět vstupních bodů průměrné zbytkové chyby<sup>2</sup>:  $\epsilon_{T_{3D}} = 5,73$  pixelů,  $\epsilon_{T_{2D}} = 12,34$  pixelů a tyto matice jsou dány následovně:

$$T_{2D} = \begin{bmatrix} -2.648 * 10 & -2.579 * 10^2 & +1.174 * 10^2 \\ -4.992 * 10 & -2.514 * 10^2 & +5.325 * 10^2 \\ -9.232 * 10^{-2} & -4.676 * 10^{-1} & 1.000 \end{bmatrix}, \quad (6.3)$$

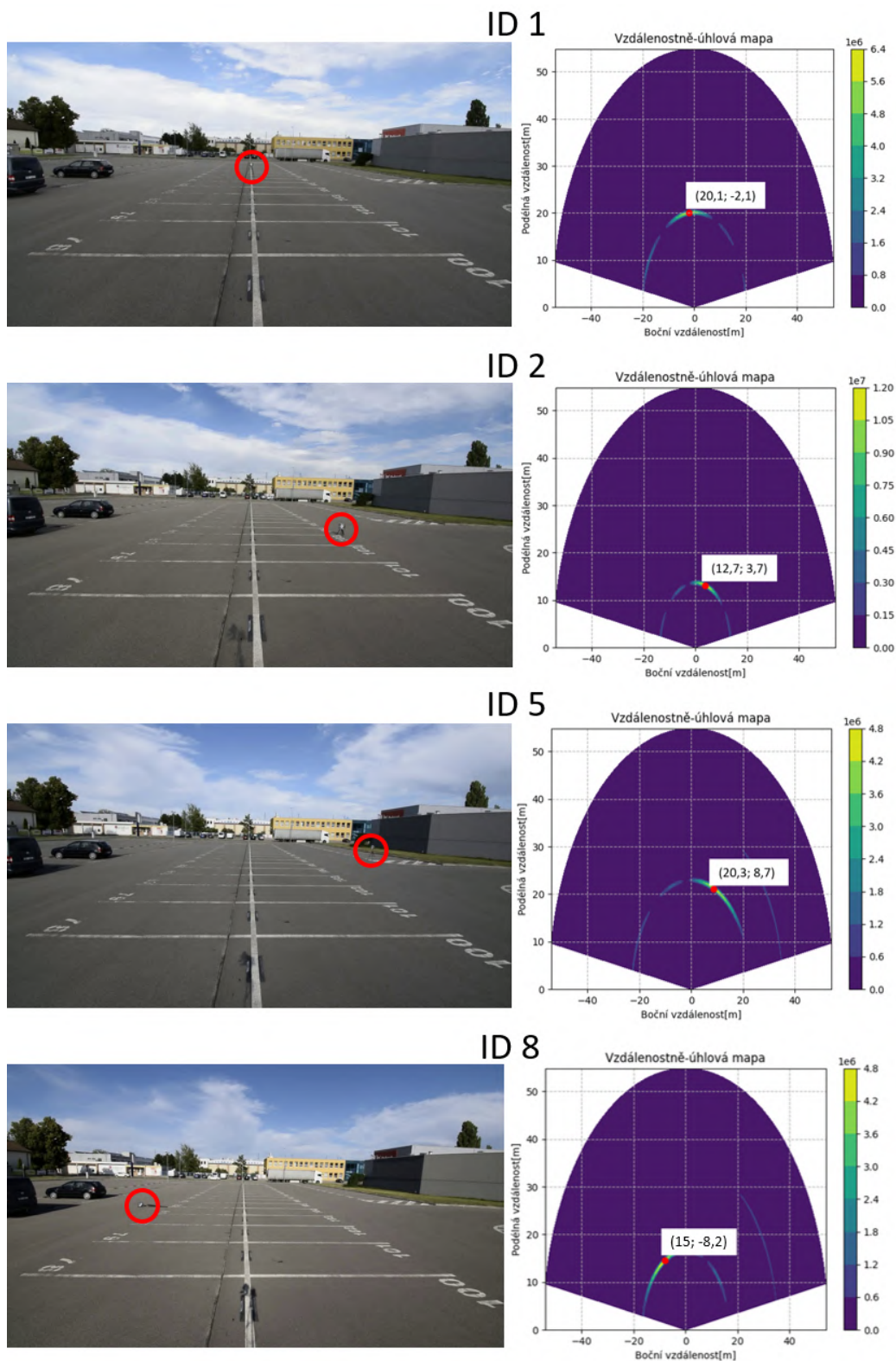
$$T_{3D} = \begin{bmatrix} +7.064 * 10^3 & +6.319 * 10^3 & +8.711 * 10^2 & +8.803 * 10^2 \\ +2.885 * 10^3 & +2.269 * 10^2 & -5.755 * 10^3 & +1.478 * 10^4 \\ +6.932 & +2.051 * 10^{-1} & +5.422 * 10^{-1} & 1.000 \end{bmatrix}. \quad (6.4)$$

V této práci jsou využita data z radaru bez informace o výšce objektů nad zemí. Je tedy třeba pro potřeby zobrazení bodů z radaru do souřadného systému kamery za pomoci 3D DLT určit konstantní výšku, v jaké mají být objekty detekované radarem zobrazeny. Při zobrazení bodů milimetrového radaru do souřadného systému kamery jsem použil nulovou výšku objektů nad zemí.

---

<sup>2</sup>Zbytková chyba představuje nepřesnost použitého modelu oproti reálnému modelovanému prostoru. Zbytková chyba je tedy způsobena redukcí mnohorozměrného prostoru na prostor modelovaný.





Obrázek 6.15: Ukázka snímků z milimetrového radaru a kamery určených pro kalibraci. Na obrázcích jsou vyznačeny pozice koutového odrazče a identifikační čísla odpovídají označení v tabulce 6.1 a tabulce 6.2.

## 6.4 Popis datové sady a předzpracování

Tato část se zaměřuje na vytvořenou a dále používanou datovou sadu obsahující data z milimetrového radaru a kamery. Pro účely testování navrženého systému jsem za pomoci prostředků popsaných v části 6.2 vytvořil vlastní datovou sadu obsahující nezpracované signály z milimetrového radaru a odpovídající snímky z kamery. Datová sada byla pořízena v ulicích města Brna a obsahuje 50 měření o délce 30s, nebo 60s a byla pořízena převážně za dobrých světelných podmínek a dobrém počasí. Datová sada však také obsahuje několik měření ve zhoršených světelných podmínkách a dvě měření byla získána v nepříznivém počasí. Pro stanovení anotací datové sady byl využit model YOLOv5-l6, který dosahuje přesnosti mean average precision (mAP) o hodnotě 71,3%. Anotace datové sady tedy vznikly automaticky.

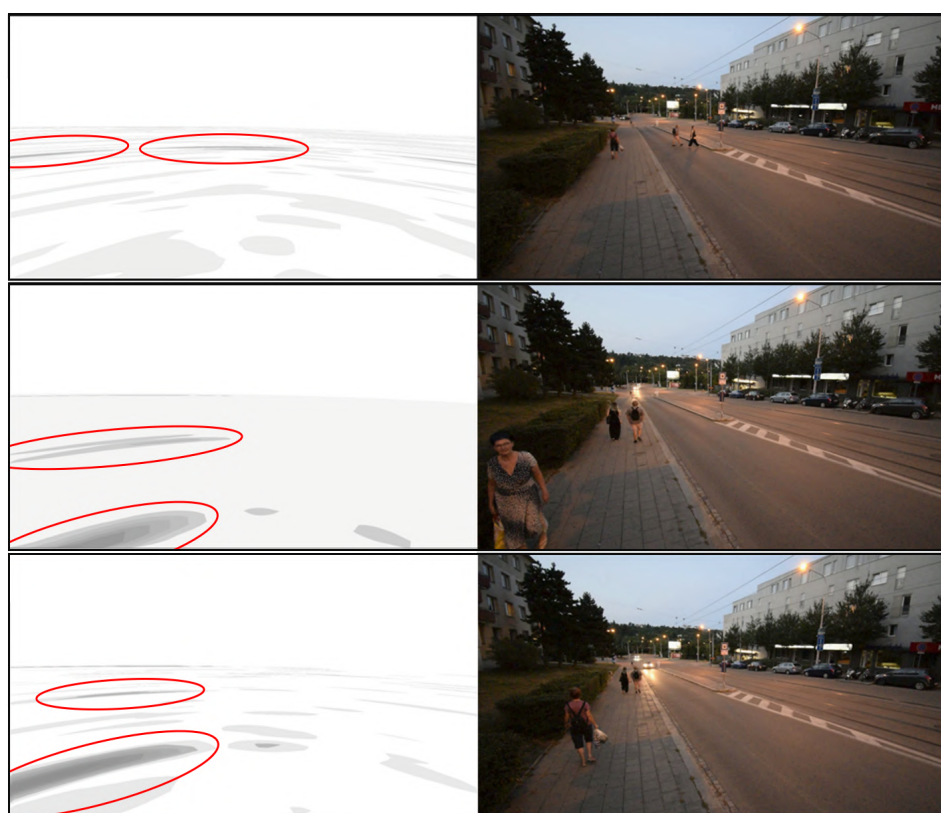
Vytvořená datová sada byla snímána s nastavením radaru daným LUA skriptem, který je zahrnut v příloze této práce. Radar zde využíval všechny tři vysílací antény společně se čtyřmi dostupnými přijímacími anténami. Je tedy z této datové sady možné určit i výšku objektu nad zemí. Další informace o použitých parametrech radaru jsou k dispozici v části 6.2, kde jsou popsány nejdůležitější parametry a rozvržení antén použitého radaru.

V rámci této práce byly pro reprezentaci nezpracovaných dat z radaru použity vzdálenostně-úhlové mapy. Vzdálenostně-úhlové mapy byly vytvořeny za pomoci zpracování vzdálenostně-rychlostně-úhlových map, které jsou výsledkem procesu popsaného v části 4. Vzdálenostně-úhlové mapy jsem vytvořil vyloučením objektů s nulovou rychlostí kvůli problémům s detekcí budov, vozovky, dopravních značek, či semaforů. Následně jsem zbylé vzdálenostně-úhlové mapy pro jednotlivé rychlosti sloučil součtem těchto rychlostních složek, objekty se všemi rychlostmi jsou tedy reprezentovány pomocí dvou-dimenzionální matice (vzdálenost, směr), dochází tím k zanedbání informace o rychlosti detekovaných objektů. Vzniklé vzdálenostně-úhlové mapy byly dále kalibrovány pomocí kalibrační metody DLT. Proces kalibrace radarových dat je popsán výše v části 6.3. Vzniklé vzdálenostně-úhlové mapy kalibrované do souřadného systému kamery jsem pak dále upravil tak, aby je bylo možné reprezentovat obrázkem a to z důvodu vhodnosti této reprezentace pro použitý fúzní systém (více viz v části 6.5) a možnosti jednoduché vizualizace. Příklady vzniklých kalibrovaných vzdálenostně-úhlových map jsou s odpovídajícími obrázky z kamery vyobrazeny na obrázku 6.16.



Obrázek 6.16: Příklady reprezentace dat z radaru a kamery.

Následuje soubor obrázků 6.17 získaných při měření ve zhoršených světelných podmínkách, kde jsou ve snímané scéně detekovány osoby. V rámci dat z radaru jsou jednotlivé osoby (viditelné na snímku z kamery) vyznačeny. Na druhém snímku v souboru obrázků 6.17 lze pozorovat směrové splynutí vzdálenějších osob kvůli nedostačující směrové rozlišitelnosti (více viz v části 4). Osoby jsou v tomto případě rozlišitelné pouze vzdálenostně.



Obrázek 6.17: Příklady reprezentace dat z radaru a kamery.

Následující soubor obrázků 6.18 vyobrazuje další příklady snímků zahrnutých v popísané datové sadě. Je zde možné opět sledovat sloučení osob detekovaných radarem, nebo jejich úplné vypuštění. Dochází zde dále k zaznamenání nežádoucího odrazu stacionárních objektů v pravé části snímků.



Obrázek 6.18: Příklady reprezentace dat z radaru a kamery.

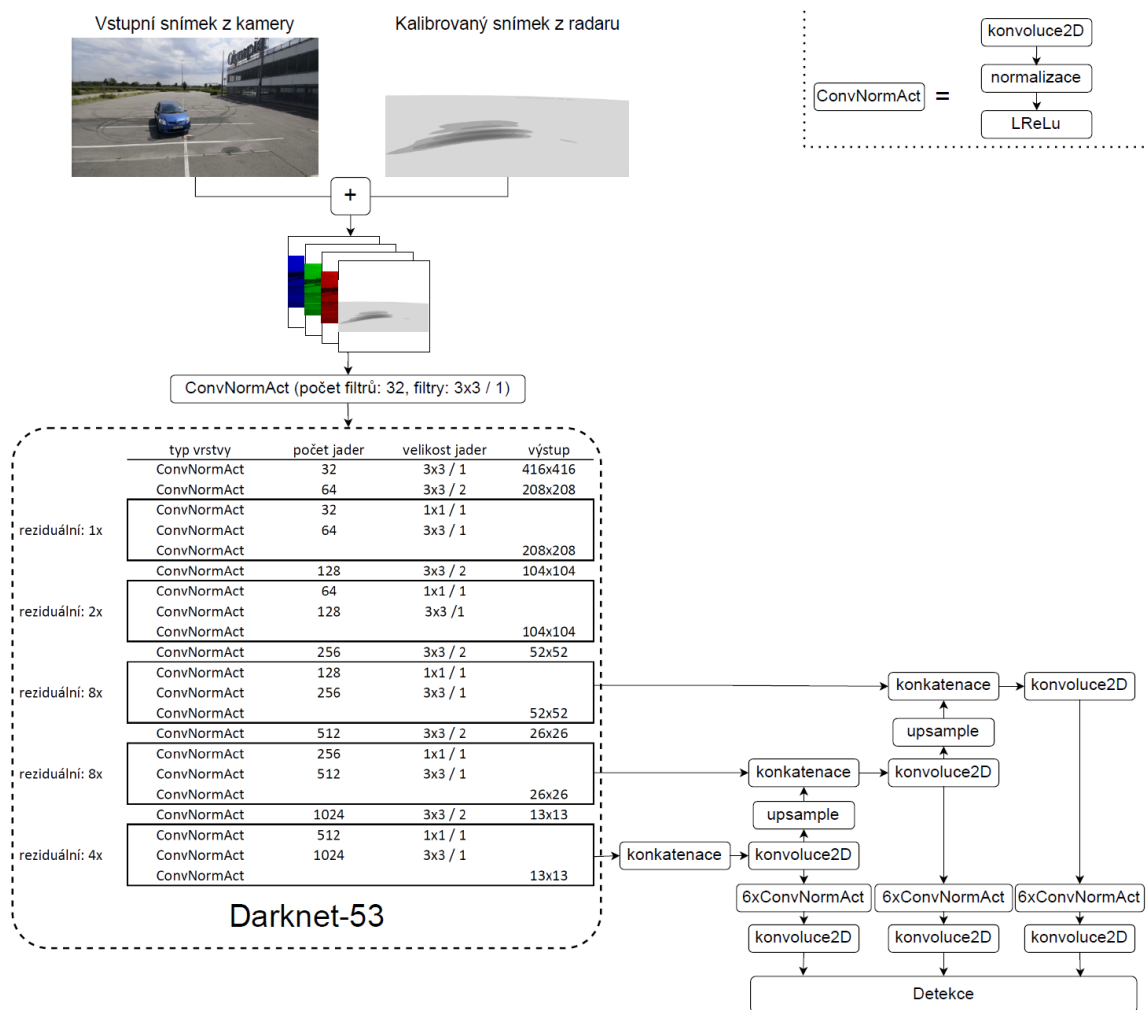
Jak je v této práci diskutováno, kamera může selhávat v nepříznivých podmínkách jako jsou počasí, smog, poletující prachové částice, či špatné světelné podmínky. Milimetrový radar však těmito podmínkami není negativně ovlivněn a může tedy pomoci kameře v překonání jmenovaných situací. Pro simulaci těchto zhoršených podmínek jsem u některých měření provedl úpravy kamerových snímků. Tyto úpravy spočívají v přidání simulace deště, mlhy, smogu, rozostření kamery a celkovém snížení jasu snímků. Zmíněné simulace byly provedeny v programu Adobe After Effects a umožňují ověřit, zda fúzní systémy mohou opravdu lépe fungovat ve zhoršených podmínkách. Příklady takto upravených snímků jsou vyobrazeny na obrázcích 6.19.



Obrázek 6.19: Příklady úprav videí pro simulaci zhoršeného počasí a špatných světelných podmínek.

## 6.5 Architektura navrženého systému

Tato část popisuje navržený fúzní systém, jehož architektura je z velké části totožná s architekturou modelu YOLOv3. Vstupem navrhovaného systému však nejsou tři vstupní kanály, nýbrž kanály čtyři. To znamená, že je ke standardnímu RGB vstupu připojen čtvrtý kanál obsahující kalibrovanou reprezentaci dat milimetrového radaru. Jedná se tedy o brzkou fúzi. Systém by takto měl nalézt vztahy mezi daty z obou sensorů v rámci procesu učení sám. Architektura systému je vyobrazena na následujícím obrázku 6.20. Architektura se od YOLOv3 liší pouze v předřazení konvoluční vrstvy s 32 filtry společně s následující normalizační a aktivační vrstvou před extraktor příznaků Darknet-53. Fúzní systém byl implementován za pomoci modulu PyTorch.



Obrázek 6.20: Architektura navrženého fúzního systému.

## 6.6 Testování navrženého systému

Tato část se zaměřuje na testování navrženého systému. Prezentuje příklady výstupů modelu, diskutuje nedokonalosti a vyhodnocuje celkovou úspěšnost systému na testovací datové sadě. Dále se zaměřuje na fungování v simulovaných nepříznivých podmínkách.

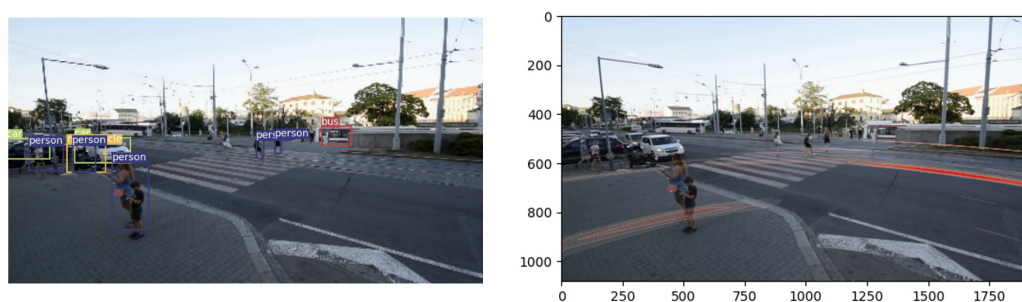
Nejprve jsou zde vyobrazeny příklady výstupů navrženého systému společně s odpovídajícími snímky z kamery obohacenými o nenulové hodnoty kalibrovaných dat z milimetrového radaru (vyobrazenými úrovněmi červené). Na prvním obrázku 6.21 je možné pozorovat, že radarem detekované osoby, stejně jako je popsáno v části 6.4, pro radarová data částečně splývají. Dalším nedostatkem pro obrázek 6.21 je detekce „false positive“ případů, kdy je chybně detekován autobus pravděpodobně kvůli detekci zítky milimetrovým radarem společně s částí střechy autobusu kamerou. Tato „false positive“ detekce se objevuje i na dalších snímcích ze stejného snímání.

Z tohoto příkladu lze tedy vyvodit, že data z radaru ovlivňují, v tomto případě nepříznivě, fungování modelu. Toto chování je dále ukázáno na obrázcích 6.26 až 6.29, kdy naopak radarová data vedou k pozitivnímu zlepšení detekce objektů. V těchto případech

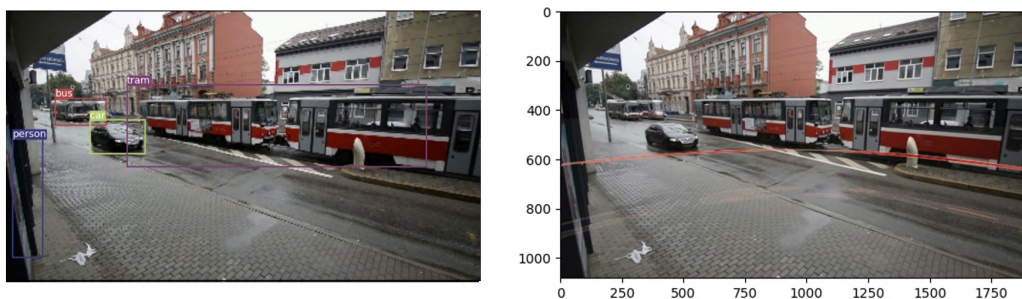
jsou snímky z kamery uměle zhoršeny a model by se měl spoléhat ve větší míře na data z radaru.



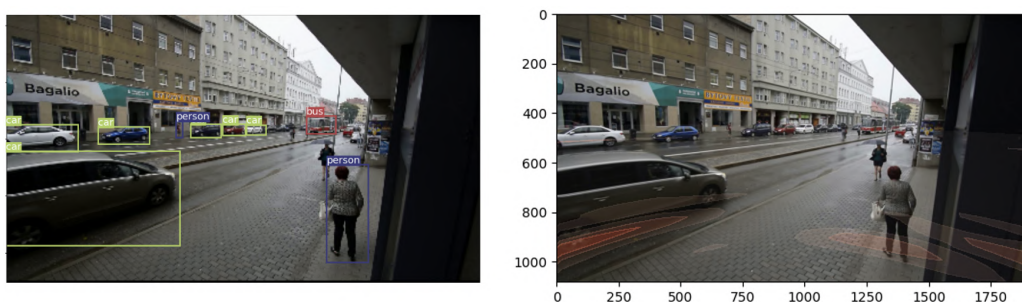
Obrázek 6.21: Výstup fúzního modelu a ukázka odpovídajících dat.



Obrázek 6.22: Výstup fúzního modelu a ukázka odpovídajících dat.

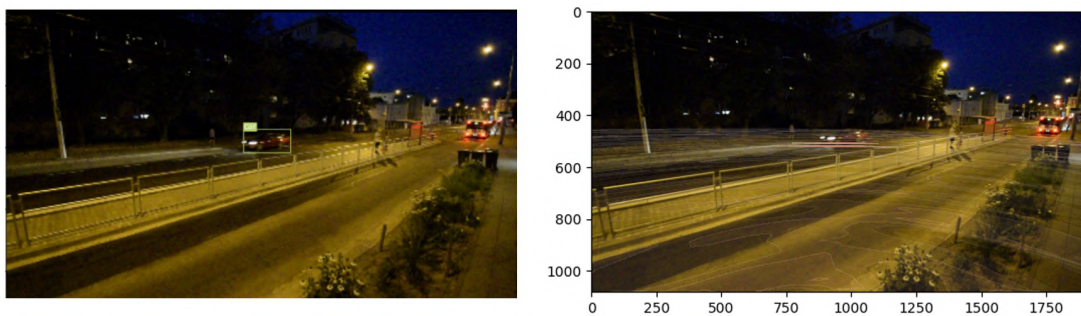


Obrázek 6.23: Výstup fúzního modelu a ukázka odpovídajících dat.

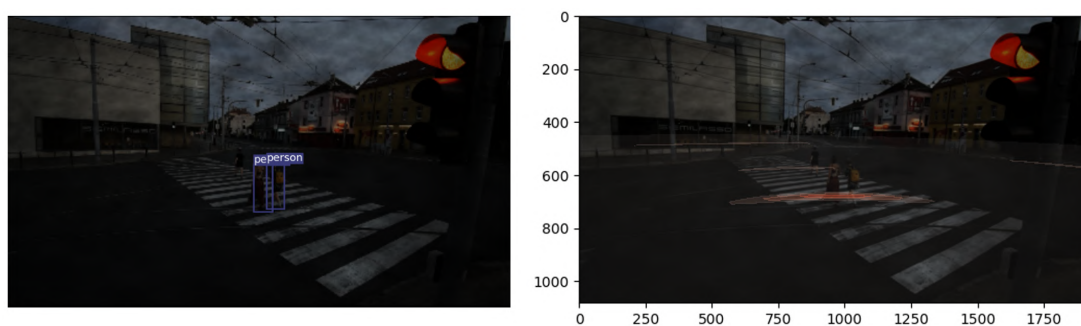


Obrázek 6.24: Výstup fúzního modelu a ukázka odpovídajících dat.

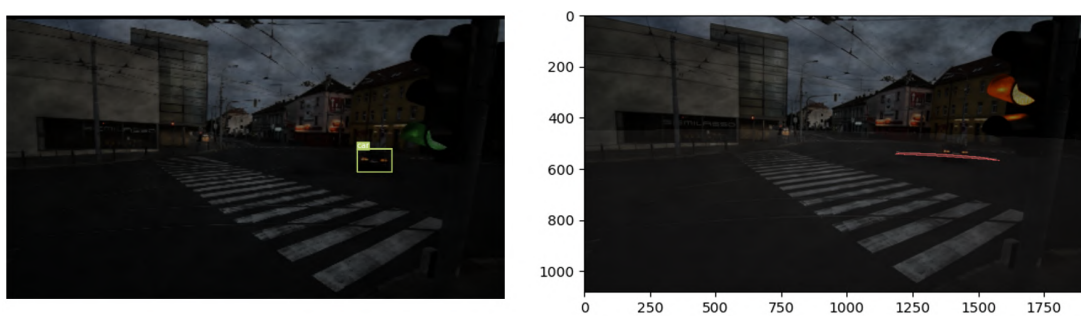
Následující obrázky ukazují, že systém dokáže fungovat i ve zhoršených podmínkách. Tyto situace byly v případě snímků 6.26 až 6.29 simulovány pro získání správných anotací (více viz v části 6.4). Obrázek 6.25 byl pořízen ve zhoršených světelných podmínkách a ukazuje, že detekční schopnost radaru není v těchto situacích zhoršena. Detekční schopnost radaru v silném dešti nebyla ověřena, kvůli voděodolnosti pro snímání použitého hardware. Jedinými dvěma případy snímání v dešti (úhrn srážek = 0,7mm/h) jsou příklady z měření reprezentované na obrázku 6.23 a obrázku 6.24.



Obrázek 6.25: Výstup fúzního modelu a ukázka odpovídajících dat.

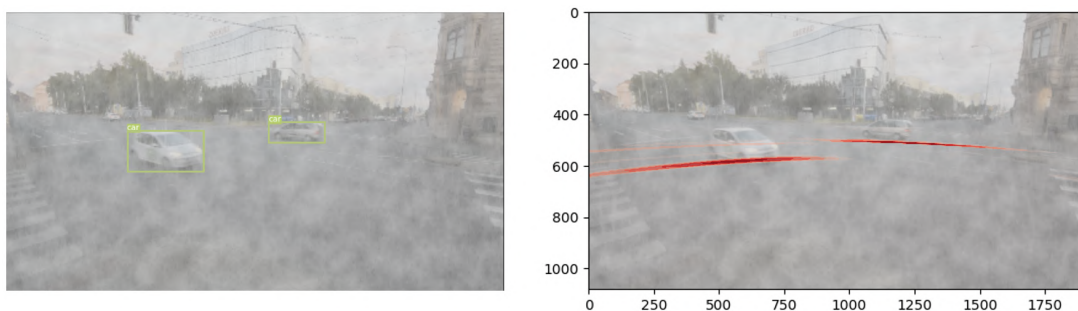


Obrázek 6.26: Výstup fúzního modelu a ukázka odpovídajících dat.

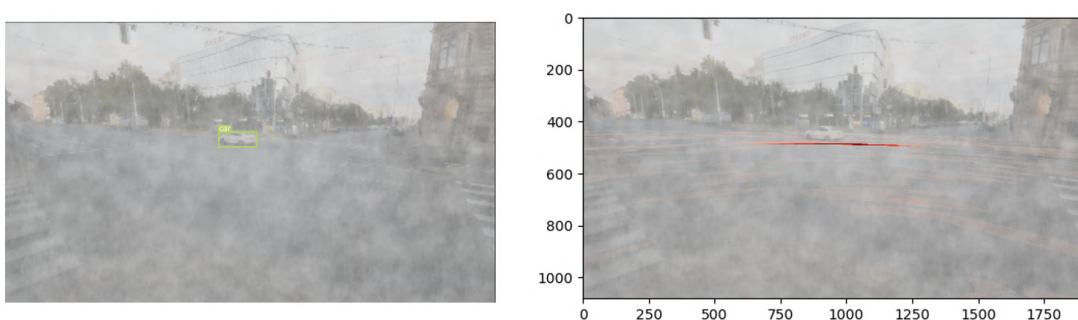


Obrázek 6.27: Výstup fúzního modelu a ukázka odpovídajících dat.





Obrázek 6.28: Výstup fúzního modelu a ukázka odpovídajících dat.



Obrázek 6.29: Výstup fúzního modelu a ukázka odpovídajících dat.

Pro testovací část datové sady bylo využito několika metrik. Tou nejdůležitější je „mean-average-precision“ (mAP), která se využívá pro ohodnocení detekčních systémů (více viz v části 2.3). Dále je pak vyhodnocena přesnost detekce objektů společně s přesností detekce pozadí a správnost klasifikace detekovaných objektů.

Testovací část datové sady, na které byly výše zmíněné metriky vyhodnoceny, se skládá z celkem sedmi měření, které dohromady tvoří záznam o délce 7m a 30s, tedy 450 snímků. Z těchto sedmi měření byly tři upraveny simulací zhoršených podmínek pro data z kamery (více viz v části 6.4).

Pro ověření detekčních schopností fúzního systému je dále uskutečněno srovnání s implementovaným a na stejné datové sadě trénovaným modelem YOLOv3. Tento model funguje samozřejmě velmi dobře pro případy v dobrých podmínkách. Příklady těchto detekcí jsou ukázány na obrázcích 6.30 až 6.33. Na těchto příkladech je možné pozorovat, že pro objekty, které jsou stacionární a nacházejí se ve větší vzdálenosti od senzorů, tento systém dokáže lépe detekovat a klasifikovat. Tento fakt lze vysvětlit tím, že radar v těchto případech dané objekty buď nedokáže detekovat kvůli maximální detekční vzdálenosti radaru, nebo se jedná o stacionární objekty, které jsou částečně filtrovány. Navržený fúzní model tedy tyto objekty detekuje s menší pravděpodobností.



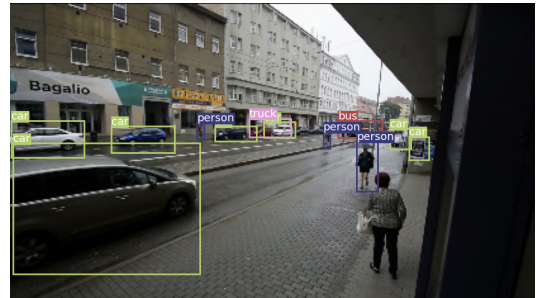
Obrázek 6.30: Výstup modelu YOLOv3.



Obrázek 6.31: Výstup modelu YOLOv3.



Obrázek 6.32: Výstup modelu YOLOv3.



Obrázek 6.33: Výstup modelu YOLOv3.

Pro situace, kde jsou snímky z kamery uměle zhoršeny, model YOLOv3 v mnoha případech selhává. Na případech 6.34 a 6.35, které reprezentují zhoršené světelné podmínky a přítomnost smogu, je možné pozorovat, že objekty nejsou správně detekovány.

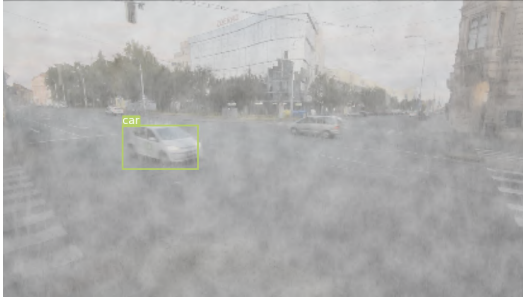


Obrázek 6.34: Výstup modelu YOLOv3.



Obrázek 6.35: Výstup modelu YOLOv3.

Dalšími příklady jsou snímky uměle zhoršené pomocí simulace deště a mlhy (snímek 6.36 a snímek 6.37). U těchto příkladů je opět možné pozorovat, že je detekční schopnost modelu YOLOv3 zhoršena a některá vozidla nejsou detekována.



Obrázek 6.36: Výstup modelu YOLOv3. Obrázek 6.37: Výstup modelu YOLOv3.

Pro vyhodnocení fúzního modelu stejně jako YOLOv3 modelu byly použity metriky popsané výše. Následující tabulka 6.3 ukazuje zjištěné hodnoty jednotlivých metrik pro oba systémy a jejich srovnání, kdy je srovnán fúzní model oproti YOLOv3.

	Fúzní model	YOLOv3	Srovnání modelů
přesnost klasifikace tříd	74,82%	76,26%	+1,44%
podíl správně detekovaných objektů	76,54%	64,70%	-11,84%
podíl správně detekovaného pozadí	99,29%	99,14%	-0,15%
mAP	0,45	0,55	+0,09

Tabulka 6.3: Tabulka vyhodnocených metrik pro fúzní model a YOLOv3 model.

Z těchto hodnot je možné vyčíst, že navržený fúzní systém ve vyšší míře detekuje objekty (pro fúzní systém vyšší hodnoty podílu správně detekovaných objektů). Nižší hodnota mAP u fúzního systému je pak dána méně přesným umístěním ohraničujícího rámečku kolem objektů (mAP vyhodnocuje IoU). Objekty jsou v těchto případech sice detekovány a správně zařazeny do odpovídající třídy, dochází však k odlišnému umístění ohraničujícího rámečku oproti anotacím. Výše popsané ohodnocení je způsobeno vyšším podílem detekcí u měření se zhoršenými podmínkami u fúzního systému oproti modelu YOLOv3.

Navržený fúzní detekční systém tedy dokáže lépe detekovat objekty v případě zhoršených podmínek díky datům z milimetrového radaru. Využití dat z radaru však dále způsobuje oproti YOLOv3 zhoršení umístění ohraničujících rámečků oproti anotacím.

Ukázalo se tedy, že brzká fúze dat milimetrového radaru a kamery může vést k detekčnímu systému, který je oproti systému spoléhajícímu se pouze na obrazová data z kamery robustnější a dokáže lépe pracovat ve zhoršených podmínkách.

Nevýhodou fúze dat z radaru a kamery pro navržený systém je zhoršení přesnosti umístění ohraničujícího rámečku pro detekované objekty. Tento problém je pravděpodobně způsoben nízkou směrovou rozlišitelností radaru, kdy je detekovaný objekt kvůli kalibraci dat vyobrazen jako širší než ve skutečnosti, což je pak viditelné na snímku z kamery. Tento fenomén je možné pozorovat ku příkladu na obrázku 6.29 nebo obrázku 6.22.

# Kapitola 7

## Závěr

Cílem této diplomové práce bylo prozkoumat různé fúzní metody využívající principy hlubokého učení a navrhnout fúzní systém. Tento záměr byl splněn.

Pro navržení vhodného fúzního systému bylo tedy třeba provést průzkum existujících fúzních metod studiemi dostupné literatury. Pro demonstraci fúze obrazových a neobrazových dat bylo pak třeba vybrat vhodnou kombinaci senzorů. Zvolenými senzory se staly pro neobrazová data milimetrový radar a pro obrazová data monokulární kamera kvůli svým komplementárním vlastnostem. Data z těchto senzorů byla dále použita pro vhodný případ ověření (monitorování dopravy) pomocí navrženého detekčního fúzního systému využívajícího konvoluční neuronové sítě. Tento systém byl pak implementován pomocí vhodných knihoven a modulů. Funkčnost systému byla dále ověřena pomocí vhodných metrik a na závěr porovnána se systémem využívajícím pouze obrazová data. Navržený fúzní systém dosahuje pro dané metriky následujících úspěšností: přesnost klasifikace tříd: 74,816%, podíl správně detekovaných objektů: 76,539% a mAP: 0,453.

V práci jsem tedy navrhl fúzní systém na základě průzkumu existujících fúzních metod spadajících do metod hlubokého učení. Tento systém byl využit pro účely detekce objektů v dopravních situacích. Dále pak byl pro účely srovnání využit detekční systém spoléhající se pouze na obrazová data a závěrem byly tyto systémy porovnány.

Práce na této diplomové práci mne naučila mnoho o zpracování signálů z milimetrového radaru a zpracování signálů pomocí Fourierovy transformace obecně. Dále se lépe orientuji v metodách hlubokého učení a především v těch zaměřených na detekci objektů a fúzi dat. Musel jsem také řešit problémy spočívající v ověření synchronizace kanálů kamery, kdy jsem musel navrhnout vhodný obvod a programovat nízkourovňový počítač Arduino. Odnáším si tedy i lepší znalosti z tohoto odvětví.

Budoucí práce by mohla být zaměřena na vytvoření průběžného fúzního systému a pozdního fúzního systému pro zjištění výhod jednotlivých typů fúze. Dalším možným pokračováním je zprovoznění navrženého fúzního algoritmu pro embedded systémy jako Nvidia Jetson nebo úprava systému pro použití v reálném čase. Dále by bylo možné pro data z milimetrového radaru získat a využít informace o umístění detekovaných objektů nad zemí pro účely přesnější kalibrace.

# Literatura

- [1] IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. *IEEE Std 1588-2002*. USA: IEEE. 2002, s. 1–154. DOI: 10.1109/IEEESTD.2002.94144. Dostupné z: <https://standards.ieee.org/ieee/1588/6825/>.
- [2] *MMWAVE STUDIO User Guide*. 02\_01\_01\_00. USA: Texas Instruments, 2016. Dostupné z: [https://software-dl.ti.com/ra-processors/esd/MMWAVE-STUDIO/02\\_01\\_00\\_00/index\\_FDS.html](https://software-dl.ti.com/ra-processors/esd/MMWAVE-STUDIO/02_01_00_00/index_FDS.html).
- [3] *60GHz mmWave Sensor EVMs*. SWRU546E. USA: Texas Instruments, 2018. Revize 2022. Dostupné z: <https://www.ti.com/product/IWR6843>.
- [4] A Definition of AI. BEL: Evropská komise. 2018, s. 1. Dostupné z: <https://digital-strategy.ec.europa.eu/en/library/definition-artificial-intelligence-main-capabilities-and-scientific-disciplines>.
- [5] ABDU, F. J., ZHANG, Y., FU, M., LI, Y. a DENG, Z. Application of Deep Learning on Millimeter-Wave Radar Signals: A Review. *Sensors*. CHE: MDPI. 2021, sv. 21, č. 6. DOI: 10.3390/s21061951. ISSN 1424-8220. Dostupné z: <https://www.mdpi.com/1424-8220/21/6/1951>.
- [6] AHMED, S., HUDA, M. N., RAJBHANDARI, S., SAHA, C., ELSHAW, M. et al. Pedestrian and Cyclist Detection and Intent Estimation for Autonomous Vehicles: A Survey. *Applied Sciences*. CHE: MDPI. 2019, sv. 9. DOI: 10.3390/app9112335. Dostupné z: <https://www.mdpi.com/2076-3417/9/11/2335>.
- [7] ASAWA, C. *CS231n: Convolutional Neural Networks for Visual Recognition* [online]. USA: Stanford University Press, 2019. Dostupné z: <http://cs231n.github.io/>.
- [8] BURKOV, A. *The Hundred-Page Machine Learning Book*. CAN: Andriy Burkov, 2019. ISBN 9781999579517. Dostupné z: <https://books.google.cz/books?id=0jbxwQEACAAJ>.
- [9] CAESAR, H., BANKITI, V., LANG, A. H., VORA, S., LIONG, V. E. et al. NuScenes: A multimodal dataset for autonomous driving. *Computer Research Repository*. USA: Cornell University. 2019, abs/1903.11027. Dostupné z: <http://arxiv.org/abs/1903.11027>.
- [10] CHANG, S., ZHANG, Y., ZHANG, F., ZHAO, X., HUANG, S. et al. Spatial Attention Fusion for Obstacle Detection Using MmWave Radar and Vision Sensor. *Sensors*. CHE: MDPI. 2020, sv. 20, č. 4. DOI: 10.3390/s20040956. ISSN 1424-8220. Dostupné z: <https://www.mdpi.com/1424-8220/20/4/956>.

- [11] DANZER, A., GRIEBEL, T., BACH, M. a DIETMAYER, K. 2D Car Detection in Radar Data with PointNets. In: *IEEE Intelligent Transportation Systems Conference 2019 (ITSC)*. NZL: IEEE, 2019, abs/1904.08414, s. 61–66. ISBN 978-1-5386-7024-8. Dostupné z: <http://arxiv.org/abs/1904.08414>.
- [12] ELHAYEK, A., STOLL, C., KIM, K. I., SEIDEL, H. P. a THEOBALT, C. Feature-Based Multi-video Synchronization with Subframe Accuracy. In: PINZ, A., POCK, T., BISCHOF, H. a LEBERL, F., ed. *Pattern Recognition*. DEU: Springer Berlin, 2012, s. 266–275. ISBN 978-3-642-32717-9. Dostupné z: [https://link.springer.com/chapter/10.1007/978-3-642-32717-9\\_27](https://link.springer.com/chapter/10.1007/978-3-642-32717-9_27).
- [13] FRIDMAN, L., BROWN, D. E., ANGELL, W., ABDIĆ, I., REIMER, B. et al. Automated synchronization of driving data using vibration and steering events. *Pattern Recognition Letters*. USA: Elsevier. 2016, sv. 75, s. 9–15. Dostupné z: <https://arxiv.org/abs/1510.06113>.
- [14] G. EL NATOUR, F. R. F. R. P. B. Radar and vision sensors calibration for outdoor 3D reconstruction. In: *Proceedings - IEEE International Conference on Robotics and Automation 2015*. USA: IEEE, 2015, č. 2015, s. 2084–2089. DOI: 10.1109/ICRA.2015.7139473. Dostupné z: [https://www.researchgate.net/publication/282951724\\_Radar\\_and\\_vision\\_sensors\\_calibration\\_for\\_outdoor\\_3D\\_reconstruction](https://www.researchgate.net/publication/282951724_Radar_and_vision_sensors_calibration_for_outdoor_3D_reconstruction).
- [15] GIRSHICK, R. Fast R-CNN. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. USA: IEEE, 2015, s. 1440–1448. DOI: 10.1109/ICCV.2015.169. Dostupné z: <https://ieeexplore.ieee.org/document/7410526>.
- [16] GIRSHICK, R. B., DONAHUE, J., DARRELL, T. a MALIK, J. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. USA: IEEE. 2014, abs/1311.2524. Dostupné z: <http://arxiv.org/abs/1311.2524>.
- [17] GUPTA, A., KOSAKA, D., PAN, E., TANG, J., YAO, R. et al. *OpenRadar: A Toolkit for Prototyping mmWave Radar Applications*. USA: Cornell University, 2019. DOI: 10.48550/ARXIV.1912.12395. Dostupné z: <https://arxiv.org/abs/1912.12395>.
- [18] HAMED HABIBI AGHDAM, E. J. H. *Guide to Convolutional Neural Networks*. 1. vyd. DEU: Springer Berlin, 2017. ISBN 9783319575490. <https://doi.org/10.1007/978-3-319-57550-6>.
- [19] HE, K., ZHANG, X., REN, S. a SUN, J. Deep Residual Learning for Image Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, 2016. Dostupné z: <http://arxiv.org/abs/1512.03385>.
- [20] HENDERSON, P. a FERRARI, V. End-to-end training of object class detectors for mean average precision. *Computer Vision – ACCV 2016*. DEU: Springer International Publishing. 2016, abs/1607.03476. Dostupné z: [https://www.researchgate.net/publication/305229075\\_End-to-end\\_training\\_of\\_object\\_class\\_detectors\\_for\\_mean\\_average\\_precision](https://www.researchgate.net/publication/305229075_End-to-end_training_of_object_class_detectors_for_mean_average_precision).
- [21] IANDOLA, F. N., MOSKEWICZ, M. W., ASHRAF, K., HAN, S., DALLY, W. J. et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model

- size. In: *5th International Conference on Learning Representations, ICLR 2017*. FRA: ICLR, 2016. Dostupné z: [https://www.researchgate.net/publication/319770256\\_SqueezeNet\\_AlexNet-level\\_accuracy\\_with\\_50x\\_fewer\\_parameters\\_and\\_textless1MB\\_model\\_size](https://www.researchgate.net/publication/319770256_SqueezeNet_AlexNet-level_accuracy_with_50x_fewer_parameters_and_textless1MB_model_size).
- [22] ISABELLE GUYON, A. E. An Introduction to Variable and Feature Selection. In: *Journal of Machine Learning*. 1. vyd. USA: Microtome Publishing, 2003, č. 3. ISSN 1532-4435. Dostupné z: [https://www.researchgate.net/publication/221996079\\_An\\_Introduction\\_of\\_Variable\\_and\\_Feature\\_Selection](https://www.researchgate.net/publication/221996079_An_Introduction_of_Variable_and_Feature_Selection).
- [23] JOHN, V., NITHILAN, M. K., MITA, S., TEHRANI, H., SUDHEESH, R. S. et al. SO-Net: Joint Semantic Segmentation and Obstacle Detection Using Deep Fusion of Monocular Camera and Radar. In: *Image and Video Technology: PSIVT 2019 International Workshops*. DEU: Springer-Verlag, 2019, s. 138–148. DOI: 10.1007/978-3-030-39770-8\_11. ISBN 978-3-030-39769-2.
- [24] JOHN, V. a MITA, S. RVNet: Deep Sensor Fusion of Monocular Camera and Radar for Image-Based Obstacle Detection in Challenging Environments. In: LEE, C., SU, Z. a SUGIMOTO, A., ed. *Image and Video Technology*. DEU: Springer International Publishing, 2019, s. 351–364. ISBN 978-3-030-34879-3.
- [25] KRIZHEVSKY, A., SUTSKEVER, I. a HINTON, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In: PEREIRA, F., BURGESS, C., BOTTOU, L. a WEINBERGER, K., ed. *Advances in Neural Information Processing Systems*. USA: MIT, 2012, sv. 25. Dostupné z: <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.
- [26] LECUN, Y., BOTTOU, L., BENGIO, Y. a HAFFNER, P. Gradient-Based Learning Applied to Document Recognition. In: *Proceedings of the IEEE*. USA: IEEE, 1998, sv. 86, č. 11, s. 2278–2324. DOI: 10.1.1.42.7665. Dostupné z: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.7665>.
- [27] LIM, T.-Y., ANSARI, A., MAJOR, B., FONTIJNE, D., HAMILTON, M. et al. Radar and Camera Early Fusion for Vehicle Detection in Advanced Driver Assistance Systems. *NeurIPS Machine Learning for Autonomous Driving Workshop*. CAN: NeurIPS. 2019. Dostupné z: <https://www.semanticscholar.org/paper/Radar-and-Camera-Early-Fusion-for-Vehicle-Detection-Lim/497dc05e1e470f8800b4888ed5e91c62be2fed8f>.
- [28] LIM, T. Y., MARKOWITZ, S. A. a DO, M. RaDICAL: A Synchronized FMCW Radar, Depth, IMU and RGB Camera Data Dataset with Low-Level FMCW Radar Signals (ROS bag format). USA: University of Illinois at Urbana-Champaign. 2021. Dostupné z: [https://doi.org/10.13012/B2IDB-3289560\\_V1](https://doi.org/10.13012/B2IDB-3289560_V1).
- [29] LIU, C., LI, Y., AO, D. a TIAN, H. Spectrum-Based Hand Gesture Recognition Using Millimeter-Wave Radar Parameter Measurements. *IEEE Access: The Multidisciplinary Open Access Journal*. POL: IEEE. 2019, sv. 7, s. 79147–79158. DOI: 10.1109/ACCESS.2019.2923122.
- [30] LIU, W., ANGUELOV, D., ERHAN, D., SZEGEDY, C., REED, S. E. et al. SSD: Single Shot MultiBox Detector. *Computer Vision – ECV 2016*. DEU: Springer International Publishing. 2016. Dostupné z: <http://arxiv.org/abs/1512.02325>.

- [31] MILLS, D. L. Internet Time Synchronization: the Network Time Protocol. *IEEE Transactions on Communications*. USA: IEEE. 1991, sv. 39, s. 1482–1493. Dostupné z: <https://systems.cs.columbia.edu/ds2-class/papers/mills-ntp.pdf>.
- [32] NABATI, R. a QI, H. RRPN: Radar Region Proposal Network for Object Detection in Autonomous Vehicles. *2019 IEEE International Conference on Image Processing (ICIP)*. TWN: IEEE. 2019, s. 3093–3097.
- [33] NGUYEN, N., LEE, S., NGUYN, T. a KIM, Y.-H. One-shot learning-based driver’s head movement identification using a millimetre-wave radar sensor. *IET Radar, Sonar Navigation*. GBR: The Institution of Engineering and Technology (IET). 2022, sv. 16. DOI: 10.1049/rsn2.12223. Dostupné z: <https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/rsn2.12223>.
- [34] OH, J., KIM, K.-S., PARK, M. a KIM, S. A Comparative Study on Camera-Radar Calibration Methods. In: *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. USA: IEEE, 2018, s. 1057–1062. DOI: 10.1109/ICARCV.2018.8581329. Dostupné z: <https://ieeexplore.ieee.org/document/8581329>.
- [35] OLIVEIRA, H., MENDONÇA, M., ALVES, R. a SOUZA, J. Camera Synchronization Using an Ausido-based Approach for Mobile Mapping Systems. In: *Proceedings from conference: The 11th International Conference in Mobile Mapping Technology*. CHN: IEEE, 2019.
- [36] RAO, S. *MIMO Radar*. SWRA554A. USA: Texas Instruments, 2017. Revize 2018. Dostupné z: [https://www.ti.com/lit/an/swra554a/swra554a.pdf?ts=1646903181788&ref\\_url=https%253A%252F%252Fwww.google.com%252F.](https://www.ti.com/lit/an/swra554a/swra554a.pdf?ts=1646903181788&ref_url=https%253A%252F%252Fwww.google.com%252F.)
- [37] REDMON, J., DIVVALA, S., GIRSHICK, R. a FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, 2016, s. 779–788. DOI: 10.1109/CVPR.2016.91.
- [38] REDMON, J. a FARHADI, A. YOLO9000: Better, Faster, Stronger. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, 2017, s. 6517–6525. DOI: 10.1109/CVPR.2017.690.
- [39] REDMON, J. a FARHADI, A. YOLOv3: An Incremental Improvement. *Computer Research Repository*. USA: Cornell University. 2018, abs/1804.02767. Dostupné z: <http://arxiv.org/abs/1804.02767>.
- [40] REICH, B. *Zpracování radarového signálu a fúze informací*. CZE, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/22650/>.
- [41] REICH, B. Exploitation of Neural Networks for Fusion of Camera and Millimeter-Wave Radar Data. In: *Proceedings of CESCg 2022*. SVK: CESCg, 2022. Dostupné z: [https://cescg.org/cescg\\_submission/exploitation-of-neural-networks-for-fusion-of-camera-and-millimeter-wave-radar-data/](https://cescg.org/cescg_submission/exploitation-of-neural-networks-for-fusion-of-camera-and-millimeter-wave-radar-data/).



- [42] REN, S., HE, K., GIRSHICK, R. a SUN, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In: CORTES, C., LAWRENCE, N., LEE, D., SUGIYAMA, M. a GARNETT, R., ed. *Advances in Neural Information Processing Systems*. CAN: NeurIPS, 2015, sv. 28. Dostupné z: <https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf>.
- [43] REZATOFIGHI, H., TSOI, N., GWAK, J., SADEGHIAN, A., REID, I. et al. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, 2019, s. 658–666. DOI: 10.1109/CVPR.2019.00075.
- [44] RUSSAKOVSKY, O., DENG, J., SU, H., KRAUSE, J., SATHEESH, S. et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*. DEU: Springer International Publishing. 2015, abs/1409.0575. Dostupné z: <https://link.springer.com/article/10.1007/s11263-015-0816-y>.
- [45] SAMUEL, A. L. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*. USA: IBM. 1959, sv. 3, č. 3, s. 210–229. DOI: 10.1147/rd.33.0210.
- [46] SIMMONS, A. a CHAPPELL, S. Artificial intelligence-definition and practice. *IEEE Journal of Oceanic Engineering*. USA: IEEE. 1988, sv. 13, č. 2, s. 14–42. DOI: 10.1109/48.551.
- [47] SIMONYAN, K. a ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. USA: Cornell University, 2014. DOI: 10.48550/ARXIV.1409.1556. Dostupné z: <https://arxiv.org/abs/1409.1556>.
- [48] SZEGEDY, C., LIU, W., JIA, Y., Sermanet, P., REED, S. et al. Going deeper with convolutions. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. USA: IEEE, 2015, s. 1–9. DOI: 10.1109/CVPR.2015.7298594.
- [49] THRUN, S., BURGARD, W. a FOX, D. *Probabilistic robotics*. Cambridge, Mass.: MIT Press, 2005. ISBN 0262201623 9780262201629. Dostupné z: <http://www.amazon.de/gp/product/0262201623/102-8479661-9831324?v=glance&n=283155&n=507846&s=books&v=glance>.
- [50] UHLÍŘ, J. *Kalibrace robotického pracoviště*. CZE, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/21997/>.
- [51] WANG, Y., JIANG, Z., GAO, X., HWANG, J., XING, G. et al. RODNet: Object Detection under Severe Conditions Using Vision-Radio Cross-Modal Supervision. *Computer Vision and Pattern Recognition*. USA: Cornell University. 2020, abs/2003.01816. Dostupné z: <https://arxiv.org/abs/2003.01816>.
- [52] YOON, H., JANG, M., HUH, J., KANG, J. a LEE, S. Multiple Sensor Synchronization with the RealSense RGB-D Camera. *Sensors*. CHE: MDPI. 2021, sv. 21, č. 18. DOI: 10.3390/s21186276. ISSN 1424-8220. Dostupné z: <https://www.mdpi.com/1424-8220/21/18/6276>.

- [53] ZHOU, Y., DONG, Y., HOU, F. a WU, J. Review on Millimeter-Wave Radar and Camera Fusion Technology. *Sustainability*. CHE: MDPI. 2022, sv. 14, č. 9. DOI: 10.3390/su14095114. ISSN 2071-1050. Dostupné z: <https://www.mdpi.com/2071-1050/14/9/5114>.