

Bakalárska práca



České
vysoké
učení technické
v Praze

FEL

Využitie realistickej simulácie vstupných vizuálnych dát pre strojové učenie

Miroslav Surák

Školiteľ: Dr. Karel Košnar, Ph.D.

Január 2019

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Surák** Jméno: **Miroslav** Osobní číslo: **453337**
Fakulta/ústav: **Fakulta elektrotechnická**
Katedra/ústav: **Katedra měření**
Studijní program: **Kybernetika a robotika**
Studijní obor: **Senzory a přístrojová technika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Využití realistické simulace vstupních vizuálních dat pro strojové učení

Název bakalářské práce anglicky:

Use of Realistic Simulation of Visual Input Data for Machine Learning

Pokyny pro vypracování:

1. Nastudujte metody pro realistickou simulaci vizuálních dat.
2. Naimplementujte metody pro tvorbu trénovacích dat (včetně ground-truth) pro vybranou metodu strojového učení.
3. Ověřte funkčnost naimplementované metody a použitelnost simulovaných dat s vybranou metodou strojového učení.
4. Proveďte a vyhodnoťte experimenty porovnávající dosažené výsledky při použití reálných a simulovaných dat v úloze strojového učení.

Seznam doporučené literatury:

- [1] John Robert Skinner, Sourav Garg, Niko Sünderhauf, Peter Corke, Ben Upcroft, Michael J Milford: High-Fidelity Simulation for Evaluating Robotic Vision Performance, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2016
- [2] Roman Sushkov: Detekce součástky a určení její polohy pro úlohu vybírání, Diplomová práce ČVUT v Praze, Fakulta elektrotechnická, katedra kybernetiky, 2017

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Karel Košnar, Ph.D., ČIIRK Praha

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **28.06.2018**

Termín odevzdání bakalářské práce: _____

Platnost zadání bakalářské práce:

do konce letního semestru 2018/2019

Ing. Karel Košnar, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

_____ Datum převzetí zadání

_____ Podpis studenta

Podakovanie

Mnohokrát ďakujem vedúcemu mojej bakalárskej práce, pánu Dr. Karlovi Košnarovi, Ph.D., za jeho ochotu a venovaný čas pri riešení danej problematiky.

Ďalej chcem poďakovať za prístup ku gridovým službám Národnej Gridovej Infraštruktúry MetaCentrum od združenia CESNET.

Za poskytnutie skriptu pre vyhodnocovanie klasifikátorov chcem poďakovať pánovi Martinovi Dörflerovi.

V poslednom rade ďakujem ČVUT, že mi je tak dobrou *alma mater*.

Prehlásenie

Prehlasujem, že som predloženú prácu vypracoval samostatne, a že som uviedol všetku použitú literatúru.

V Prahe, 8. januára 2019

Abstrakt

V tejto bakalárskej práci bolo predstavené použitie syntetických dát pre skvalitnenie učenia neurónových sietí. Na príklade zistenia polohy a orientácie objektu pre potreby jeho uchopenia a zdvihnutia bol vytvorený v programe POV-Ray model skutočného objektu. Pomocou tohto modelu bolo vygenerovaných 10 000 umelých snímok ako tréningová množina. Bolo otestovaných 10 klasifikátorov naučených len na umelých dátach a otestované na reálnych. V priemere, je oproti klasifikátoru naučenom na reálnych snímkoch výsledok horší len o 4%. Pri kombinácii reálnych a umelých snímok bolo pozorované zlepšenie klasifikátoru o 1.5%.

Kľúčové slová: neurónová sieť, ray tracing, syntetické dáta, zdvíhanie objektov

Školiteľ: Dr. Karel Košnar, Ph.D.

Abstract

This bachelor thesis introduces a usage of synthetic dataset to improve the training of neural networks. The usage of convolutional neural network for determination of position and orientation of an object in the task of bin picking is used as a reference task. Dataset consisting of 10 000 synthetic images of metallic disk was generated in the raytracing software. Ten different classifiers were trained making use different size of synthetic dataset and combination with real images. The performance and precision of these classifiers were tested on real inputs and compared to classifier trained on 1000 real images. The precision of classifiers using only the synthetic dataset was only 4% worse and classifiers using the combination of real and synthetic data were even 1.5% better in average.

Keywords: neural network, ray tracing, binpicking, synthetic data

Title translation: Use of Realistic Simulation of Visual Input Data for Machine Learning

Obsah

1 Predstavenie	1	3 Príprava datasetu a použitá neurónová sieť	13
1.1 Motivácia	1	3.1 Príprava datasetu	13
1.2 Súvisiace práce	2	3.1.1 Detekcia polohy a orientácie objektu	17
1.2.1 Využívanie umelých dát	2	4 Praktická časť	19
1.2.2 Rozpoznávanie polohy objektu	2	4.1 Experimenty a výsledky	19
2 Neurónové siete	5	5 Záver	31
2.1 Úvod	5	Literatúra	33
2.2 História	5	A Obsah priloženého CD	37
2.3 Neurón	7		
2.3.1 Aktvivačné funkcie	8		
2.4 Architektúry neurónových sietí .	10		
2.4.1 Dopredné siete	10		
2.4.2 Spätnoväzobné siete	11		
2.5 Učenie neurónových sietí	12		

Obrázky

2.1 Perceptron	6	4.3 Graf výsledku učenia na reálnych dátach č. 3	21
2.2 Model neurónu	8	4.4 Graf výsledku učenia na reálnych dátach č. 4	21
2.3 Sigmoid	9	4.5 Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych	22
2.4 tanh	9	4.6 Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych	22
2.5 ReLU	10	4.7 Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych	23
2.6 Dopredná sieť	11	4.8 Priemerné odchylky x a y zložiek polohy objektu pri učení na umelých snímkoch	24
2.7 Spätoväzobná sieť	12	4.9 Medián odchýliek x a y zložiek polohy pri učení na umelých snímkoch	25
3.1 Snímok z robota	14	4.10 Priemerné odchylky a ich medián z zložky polohy pri učení na umelých snímkoch	25
3.2 Technický výkres súčiastky	14		
3.3 Porovnanie súradnicových sústav robota a POV-Ray	15		
3.4 Vytvorený model	17		
4.1 Graf výsledku učenia na reálnych dátach č. 1	20		
4.2 Graf výsledku učenia na reálnych dátach č. 2	20		

4.11 Koeficienty predpovede x a y zložiek polohy normálového vektoru pri učení na umelých snímkoch ...	26	4.19 Graf výsledku učenia na kombinácií reálnych a umelých dát č. 4	30
4.12 Priemerné odchyľky x a y zložiek polohy pri učení na umelých a reálnych snímkoch	26		
4.13 Medián odchyľiek x a y zložiek polohy pri učení na umelých a reálnych snímkoch	27		
4.14 Priemerné odchyľky a ich medián z zložky polohy pri učení na umelých a reálnych snímkoch.....	27		
4.15 Koeficienty predpovede x a y zložiek polohy normálového vektoru pri učení na umelých a reálnych snímkoch	28		
4.16 Graf výsledku učenia na kombinácií reálnych a umelých dát č. 1	28		
4.17 Graf výsledku učenia na kombinácií reálnych a umelých dát č. 2	29		
4.18 Graf výsledku učenia na kombinácií reálnych a umelých dát č. 3	29		



Kapitola 1

Predstavenie



1.1 Motivácia

V posledných dekádach dochádza v priemysle k vysokej miere automatizácie. Využívanie robotov v činnostiach, ako zvarovanie či jednoduché premiestňovanie predmetov je už bežnou praxou. Príkladom je zdvíhanie objektov, kde je poloha a orientácia objektu deterministická. Problém nastáva, keď dôjde k zmene jednej z daných vlastností (napr. pri prevoze súčiastok). Tento problém je možné riešiť využívaním inteligentných sensorov a robotov, ktorí dokážu pomocou neurónových sietí určiť polohu a orientáciu objektu len s pomocou sensorických dát dostupných samotnému robotovi. Túto sieť je samozrejme nutné naučiť na veľkom množstve tréningových dát. V prípade reálnych tréningových dát je potrebné ich manuálne zozbierať a riadne kategorizovať. V reálnom prostredí je kvôli meniacim sa okolitým podmienkam, ako napríklad svetlo, pohybujúce sa okolité predmety a iné, takéto dáta zozbierať veľmi náročné. Ako sa uvádza v tejto práci [2], je pre konvolučnú neurónovú sieť (CNN) potrebné 50 tisíc snímok a viac ako 700 hodín manuálnej práce zberu dát, aby sa zabránilo vážnym nepresnostiam pri testovaní na nových, nekategorizovaných snímkoch.

Z toho vyplýva, že má zmysel skúmať rôzne spôsoby zrýchlenia, nie len učenia

sietí, ale aj zberu tréningových dát. Dnes je k dispozícii viacero spôsobov generovania umelých dát, ale príliš veľa umelých snímok môže opäť viesť na overfitting. Je preto nutné zvoliť metódy generovania umelých snímok, čo najreálnejším spôsobom. Jeden z nich bol zvolený aj pre túto prácu.

1.2 Súvisiace práce

1.2.1 Využívanie umelých dát

Používanie syntetických dát na učenie neurónových sietí nie je úplná novinka. Príkladom je práca [21], kde rozšírením tréningovej množiny dát o 3D CAD modely pre hlbokú konvolučnú neurónovú sieť dosiahli efektívnejšie výsledky v rozpoznávaní objektov. Hlavne v prípade, keď sú skutočné dáta obmedzené alebo nedostatočne klasifikované. Aj keď mali CAD modely nedostatky v oblasti textúr, polohy alebo pozadia, tak výrazne zlepšili rozpoznávanie objektov, ktorým práve tieto detaily chýbali.

Ďalším príkladom je [19], kde využili hlavné výhody simulovaných dát, a to úplnú kontrolu nad svetelnými a atmosférickými podmienkami, v oblasti rozpoznávania miesta, SLAM a rozpoznávania objektov. Využili foto-realistický gaming engine k simulovaniu prostredia a objektov z rôznych uhlov pohľadu a rôznymi svetelnými podmienkami.

V [20] zase vyvinuli rýchly a prispôsobiteľný generátor snímok textu prirodzene umiestneného v reálnom prostredí. Pomocou takýchto snímok dokázali naučiť plne konvolučnú regresnú sieť detekovať a lokalizovať text a výrazne prekonali súčasné metódy rozpoznávania textu v snímkoch.

1.2.2 Rozpoznávanie polohy objektu

Staršie princípy zdvíhania objektov vychádzajú z predpokladu, že objekty sú uložené systematicky, v radoch alebo na dopredu známych miestach. Tento

spôsob samozrejme nie je robustný a v dnešnej dobe nedostačujúci. V oblasti rozpoznávania pozície a polohy bolo do súčasnosti predstavených viacero konceptov. V [9] z roku 1987 bol predstavený systém, ktorý rozoznáva z hromady objektov jednotlivé kusy a určí niektoré dôležité informácie, ako napríklad, ktorý objekt je na vrchu. Princípom je využívanie 3D modelov objektov na ich nájdenie v daných dátach. K tomu využíva metódy porovnávania výrazných hrán a postupným hľadaním ďalších zhodných prvkov objektu.

Jeden z modernejších prístupov k tejto problematike je popísaný v [11], kde je použitá multi-flash kamera, ktorá najprv zachytí robustné hrany a následne je použitý algoritmus, ktorý spoľahlivo detekuje objekty a stanoví ich orientáciu. Ďalšie riešenie tejto problematiky už využívajúce neurónové siete je napr. [5], kde pomocou CNN dokázali spoľahlivo určiť 6D polohu objektu v rôznorodom prostredí, a to tak, že zosnímali a kategorizovali prostredie z rôznych pohľadov, do ktorých následne vkladali dopredu naskenovaný 3D model objektu. V ďalšej práci [4] je prezentovaná neurónová sieť pre detekovanie stabilného uchopenia objektu ľudskou rukou len z čiastočných snímok objektu. K tomu využíva RGBD obrázky polohy prstov a dlane na známom modeli objektu, ktoré boli vygenerované simulátorom uchopenia.



Kapitola 2

Neurónové siete



2.1 Úvod

Táto kapitola sa zameriava na oboznámenie čitateľa so strojovým učením a neurónovými sieťami. Na začiatku je rozobratá história vývoja neurónových sietí, ďalej sa venuje základným princípom fungovania neurónu, najčastejším aktivačným funkciám a deleniu neurónových sietí.



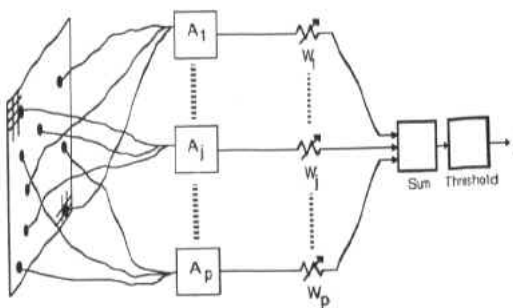
2.2 História

Simulácie neurónových sietí sa zdajú byť súčasným vývojom, no pravdou je, že táto oblasť počítačovej vedy tu bola už pred príchodom počítačov. Mnoho dôležitých pokrokov bolo dosiahnutých vďaka lacnej počítačovej sile v relatívne blízkej minulosti.

Prvý umelý model neurónu navrhli neurovedec Warren McCulloch a logik Walter Pitts (1943), ktorí svoj vývoj postavili na ich chápaní neurológie. Ich model bol založený na princípe binárneho prístroja a výsledkom boli

logické funkcie ako disjunkcia a konjunkcia. Ďalšie pokusy boli už s použitím počítačových simulácií. Výskumníci IBM (Farley a Clark, 1954) boli schopní udržovať kontakt s neurovedcami na McGill University a tak mohli svoje modely konzultovať. Táto interakcia zaviedla do oboru multidisciplinaritu, ktorá pretrváva dodnes.

Okrem neurovedy, zohrali pri vývoji veľkú rolu aj psychológovia a inžinieri. Rosenblatt, psychológ, ktorý v roku 1958 predstavil perceptrón, ktorý New York Times označil za “embryo elektronických počítačov, od ktorého americká armáda očakáva, že bude chodiť, rozprávať, vidieť, písať, reprodukovať sa a bude vedomé svojej existencie”. [24]



Obrázok 2.1: Perceptrón

Ďalším príkladom je systém ADALINE (ADAPTIVE LINEAR NEURON) (1960). Bol to analógový prístroj, ktorý implementoval jednovrstvú neurónovú sieť.

V roku 1969 vo svojej knihe Minsky a Papert generalizovali limity jednovrstvého perceptrónu voči viacvrstvovému systému. Dôsledkom čoho bolo zníženie záujmu o financovanie a naštartovali vlnu predsudkov voči tejto oblasti výskumu.

Napriek minimálnemu financovaniu a malému verejnemu záujmu viacerí výskumníci pokračovali vo vývoji metód využívajúcich neurónové siete na riešenie problémov ako napríklad rozpoznávanie vzorov. V tomto období vzniklo viacero foriem, ktoré boli dôležité pre ďalší vývoj tohto odvetvia. Príkladom sú ART (Adaptive Resonance Theory) siete (Grossberg a Carpenter, 1988), Werbos (Paul Werbos, 1974) vyvinul a použil metódu spätnej propagácie, dnes asi najviac používaná metóda v učení neurónových sieti. Amari (A.

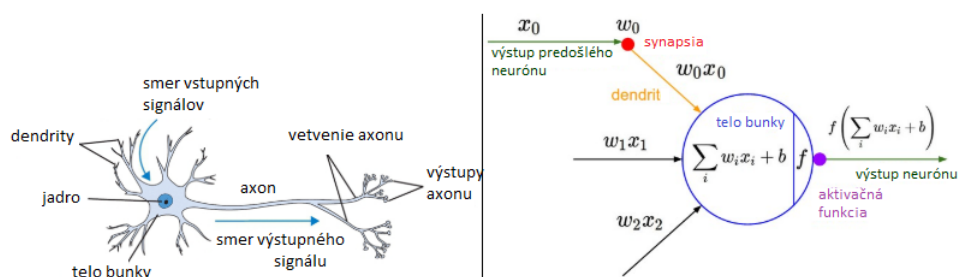
Shun-Ichi, 1967) publikoval prácu, ktorá položila základy pre teóriu učenia neurónových sietí (error-correction method).

Tento progres spôsobil dôležité znovuoživenie záujmu o neurónové siete a ich vývoj. Začali sa organizovať konferencie, médiá začali venovať oblasti viac pozornosti a publikovať návody, ktoré pomohli ešte viac rozšíriť neurónové siete. Akademické programy a kurzy sa objavili na významných technických univerzitách v USA a EU.

V dnešnej dobe už vidíme dôsledky tohto veľkého pokroku. Veľký komerčný záujem spôsobil vývoj neurónových sietí v mnohých oblastiach každodenného života.

2.3 Neurón

Neurón je základným výpočtovým prvkom mozgu. Na obrázku 2.2 je vidieť podobnosť skutočného, biologického neurónu a matematického modelu [22]. Každý neurón prijíma vstupné signály svojimi dendritmi (vstupný nervový výbežok) a produkuje výstupný signál na svojom axone (výstupný nervový výbežok). Ten sa rozdeľuje a šíri signál na dendridy iných neurónov. V matematickom modeli signál (x_0) na výstupe neurónu postupuje na vstupy ďalších neurónov vynásobený o váhu (ω_0) danej synapsie (prepojenie). Princípom váhy synapsie je kontrola vplyvu neurónu na ďalší neurón. V základnom modeli, dendridy prenášajú signál do tela bunky, kde sa prichádzajúce signály sčítavajú a po dosiahnutí hraničnej/prahovej (threshold) hodnoty, neurón vyšle signál na svoj výstup. Na základe frekvencie vysielania výstupného signálu neurónu modelujeme aktivačnú funkciu neurónu.



Obrázok 2.2: Model neurónu

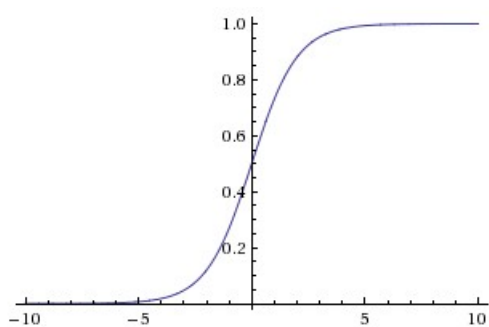
2.3.1 Aktivačné funkcie

Aktivačných funkcií je mnoho, preto sú v tejto práci popísané len tie často používané, či už v minulosti alebo v súčasnosti.

Sigmoid

$$\sigma(x) = \frac{1}{(1 + e^{-x})}$$

Dnes už síce sporadicky používaná aktivačná funkcia bola v minulosti využívaná kvôli svojej jednoduchej interpretácii. Ako je už vidieť z obrázku 2.3, funkcia zobrazí reálnu hodnotu signálu do intervalu $\langle 0, 1 \rangle$, kde veľkým záporným hodnotám priradí hodnotu 0 a veľkým kladným hodnotám 1. V súčasnosti sa sigmoid nepoužíva kvôli jeho saturácii, a teda takmer nulovému gradientu a kvôli tomu, že jeho výstup nie je centrován na nulu.

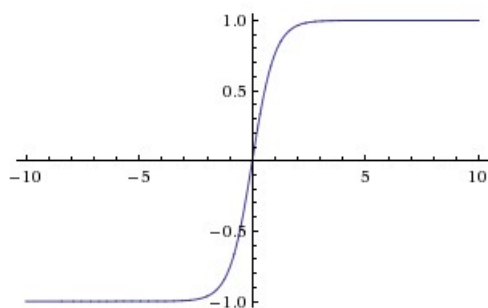


Obrázok 2.3: Sigmoid

■ Tanh

$$\tanh(x) = 2\sigma(2x) - 1$$

Tanh je nelineárna funkcia, ktorá zobrazuje vstupné hodnoty do intervalu $\langle -1, 1 \rangle$. Podobne ako sigmoid, saturuje, ale jej výstup je centrovaný na stred 2.4



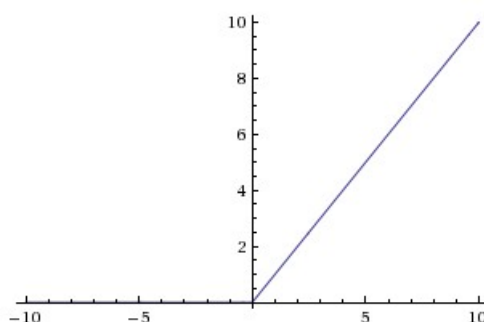
Obrázok 2.4: tanh

■ ReLU

$$f(x) = \max(0, x)$$

ReLU (Rectified linear unit) dnes veľmi populárna aktivačná funkcia. Minimálna možná výstupná hodnota tejto funkcie je 0. Oproti sigmoidu/tanh

funkcii neobsahuje náročné operácie a preto je jednoduchšia na implementáciu. Nevýhodou je často sa vyskytujúci jav takzvaného "zomierania" neurónov. Ten nastane v momente veľkého gradientu prechádzajúceho skrz neurón a spôsobí zmenu váhy takým spôsobom, že neurón už nebude aktívny pri žiadnom vstupe. Tento problém sa snaží riešiť podobná aktivačná funkcia Leaky ReLU, ktorá pri negatívnom vstupe namiesto nulovej hodnoty priradí hodnotu veľmi malú, ale kladnú (napr. 0,01), jej výsledky ale nie sú vždy konzistentné.



Obrázok 2.5: ReLU

■ Maxout

Maxout je funkcia, ktorá generalizuje ReLU a Leaky ReLU. Jej výstupom je maximálna hodnota vstupov.

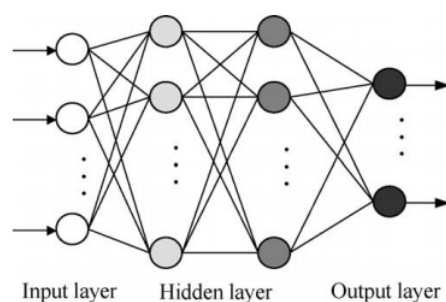
$$f(x) = \max(w_1^T x + b_1, w_2^T x + b_2)$$

■ 2.4 Architektúry neurónových sietí

■ 2.4.1 Dopredné siete

Takzvané Feed-forward networks, umožňujú signálom putovať len jedným smerom, zo vstupu na výstup. Takáto sieť je tvorená väčšinou viacerými

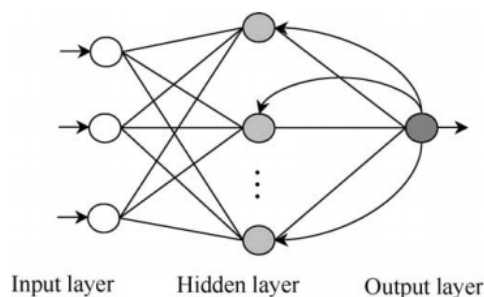
vrstvami [Obr. 2.6], kde je každý neurón danej vrstvy prepojený s každým neurónom ďalšej vrstvy. Nie je tu žiadna spätná väzba, a teda výstup konkrétnej vrstvy ju samotnú neovplyvní. Každý neurón v sieti rieši len časť celého problému. Výsledok je tvorený kompozíciou príspevkov jednotlivých neurónov. Tieto typy sietí sa často používajú pre rozpoznávanie vzorov. Najčastejším typom je Convolutional, teda konvolučné dopredná neurónová sieť.



Obrázok 2.6: Dopredná sieť

■ 2.4.2 Spätnoväzobné siete

Takzvané Feedback networks, taktiež nazývané recurrent sú siete, ktoré obsahujú spätnoväzobné prepojenia neurónov. Tieto prepojenia umožňujú signálom prechádzať v oboch smeroch. Takéto siete sú veľmi výkonné, no zároveň dokážu byť veľmi komplikované. Spätnoväzobné siete sú dynamické, teda ich stav sa mení plynulo, kým nedosiahnu stav ekvilibria. V ňom zotrývajú, kým sa nezmení vstup siete a je nutné nájsť nové ekvilibrium. Spätnoväzobné siete teda umožňujú robiť predbežné predpovede výsledku. Veľkou výhodou je, že tieto siete vedia využívať interné stavy ako pamäť, čo je dôležité pri uvedomovaní si kontextu. Ich hlavným využitím je rozpoznávanie reči, písma a textu.



Obrázok 2.7: Spätnoväzobná sieť

2.5 Učenie neurónových sietí

Všetky metódy učenia neurónových sietí sa dajú kategorizovať do dvoch hlavných skupín, a to s učiteľom alebo bez.

V prípade učenia bez učiteľa sú poskytnuté len vstupné dáta bez príslušných požadovaných výstupných informácií. Takéto siete majú za úlohu samy nájsť štruktúru daných dát. V tejto práci je využívané učenie s učiteľom, čo je bežnejší spôsob učenia, preto bude bližšie rozpísaný tento typ.

Často používaným algoritmom pre učenie s učiteľom je algoritmus spätnej propagácie. Spočíva v prepočítavaní váh siete podľa odchýlky od požadovaného výsledku. Odchýlku počíta chybová funkcia. Úlohou učenia neurónovej siete je minimalizovať hodnotu chybovej funkcie, prípadne doceliť jej nulový výstup. Na začiatku učenia sa nastavujú požadované parametre a následne sa vypočíta smer, ktorým sa znižuje výsledok chybovej funkcie.

Cieľom je nájsť také parametre, aby chybová funkcia bola v globálnom minime. K tomu, aby sa funkcia nezasekla na lokálnom minime, sa využíva niekoľko algoritmov (napr. SGD, RMSProp). Aby sa presnosť učenia ešte zlepšila, spracováva sieť vstupné hodnoty viackrát. Každé spracovanie všetkých vstupných hodnôt sa nazýva epocha. Učenie prebieha určený počet epoch alebo dokiaľ nie je splnená podmienka, že hodnota chybovej funkcie sa už niekoľko epoch nezmenila, alebo sa zmenila málo či k horšiemu.

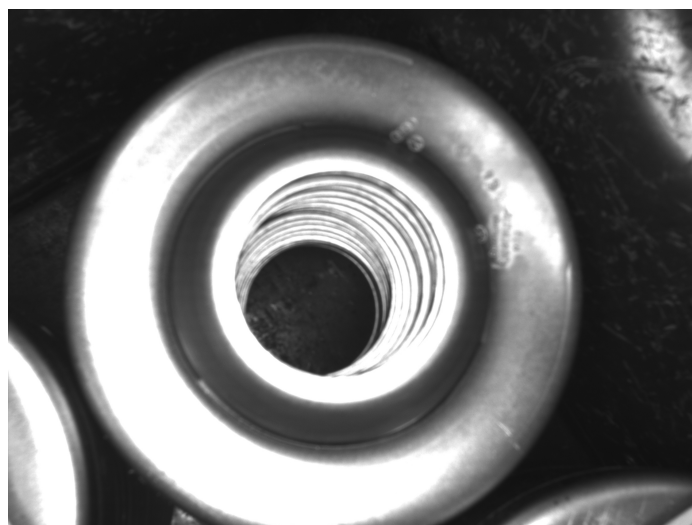
Kapitola 3

Príprava datasetu a použitá neurónová sieť

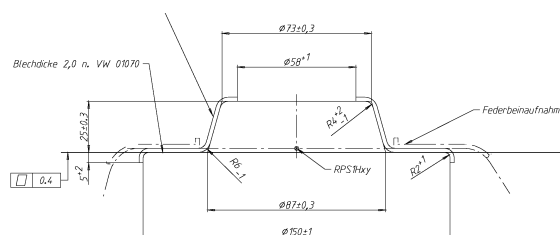
V tejto kapitole bude popísaná tvorba datasetu, popis zvoleného testovacieho objektu a použitá neurónová sieť.

3.1 Príprava datasetu

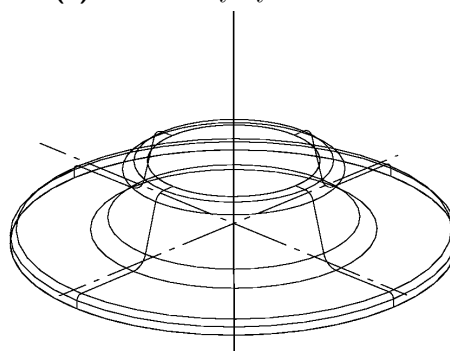
Ako testovací objekt bol zvolený hliníkový disk, slúžiaci ako držiak pružiny tlmiča od spoločnosti Škoda Auto a.s. 3.1 Jedná sa o kruhový objekt, s priemerom približne 15 cm a jeho tvar pripomína otočený tanier s vyrezaným dnom. Objekt bol poskytnutý aj s technickým výkresom 3.2, ktorý pomohol pri jeho modelovaní. Pri porovnávaní modelu podľa technického výkresu, boli pozorované drobné rozdiely, ktoré boli po manuálnom domeraní skutočného objektu doladené.



Obrázok 3.1: Snímok z robota



(a) : Technický výkres 1



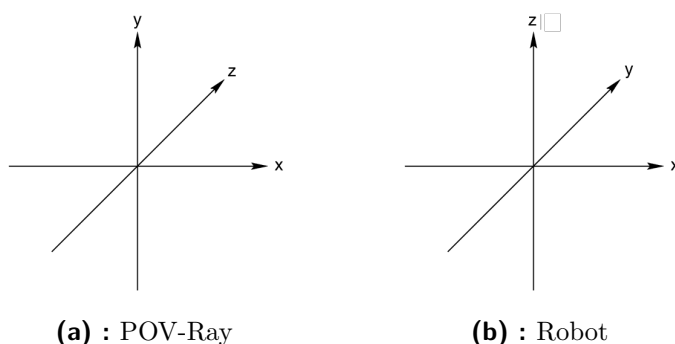
(b) : Technický výkres 2

Obrázok 3.2: Technický výkres súčiastky

Model testovacieho objektu musí čo najviac zodpovedať skutočným snímkom. Tie sú v originále v rozlíšení 2 592 x 1 944 v grayscale, kde každý pixel

obsahuje hodnotu intenzity svetla. Výber vhodného nástroja pre generovanie takýchto dát bol veľmi dôležitý. Ten musí zvládnuť generovať snímky s reálnym simulovaním svetla, textúr a snímky v grayscale. Ako najvhodnejšou voľbou sa zdá byť program, ktorý využíva ray tracing, metódu vykresľovania obrázkov stopovaním svetelných lúčov a simulovaním skutočných odrazov svetla od objektov. Zvolený program je voľne dostupný POV-Ray (Persistence of Vision Raytracer)[18], ktorý požadované funkcie umožňuje. Kód pre modelovanie testovaného objektu je v prílohe [okaz] Ako svetelný zdroj slúži 12 vymodelovaných LED diód, ktoré sa nachádzajú v okolí kamery a tak simulujú reálne rameno robota, ktorým boli skutočné snímky nasnímané. POV-Ray umožňuje výber veľkého množstva materiálov, textúr a povrchových úprav (odrazivosť svetla, rozptyl, pigment, vzory, atd.). Tie bolo potrebné nastaviť a doladiť tak, aby sa model čo najviac podobal reálnemu objektu. Taktiež je vhodné, aby tieto parametre nebolo nutné meniť s meniacou sa vzdialenosťou a orientáciou objektu.

POV-Ray a robot využívajú odlišné súradnicové sústavy. Z porovnania na obrázku 3.3 je vidieť, že stačí v programe POV-Ray objekt otočiť okolo osi x o 90 stupňov podľa pravidla pravej ruky.



Obrázok 3.3: Porovnanie súradnicových sústav robota a POV-Ray

Následne máme len opačný smer osi y , preto je nutné snímok vygenerovaný programom POV-Ray horizontálne obrátiť. Vykreslenie jedného snímku v rozlíšení 800x600 pixelov trvá približne 4 minúty. Pri veľkom množstve snímok tak vzniká problém časovej náročnosti, ktorý bol vyriešený využitím gridu. Konkrétne boli využité služby MetaCentra [23], projekt združenia CESNET, kde bolo možné si vyžiadať prístup bezplatne na základe štúdia na

ČVUT.

Metacentrum je distribuovaná výpočtová infraštruktúra a dátové úložisko, ktoré umožňuje efektívne a spoľahlivo vykonávať veľmi náročné výpočtové úlohy. Existujú dva spôsoby ako využiť gridové služby. Vytvorením interaktívneho prostredia a manuálnym ovládaním výpočtu alebo vytvorením bash skriptu. Pre účely tejto práce je postačujúce vytvorenie bash skriptu [job.sh], ktorý si vyžiada potrebné prostriedky, ako počet jadier procesoru, operačnú pamäť, či približný časový odhad úlohy, nakopíruje potrebné súbory z lokálneho úložiska na tzv. uzol. Potom vyžiada potrebné moduly z dostupnej ponuky a spustí vykresľovanie. Po skončení skopíruje snímky späť na lokálne úložisko. Týmto spôsobom je možné spustiť viac úloh súčasne a výrazne skrátiť dobu tvorby datasetu. V mojom prípade som schopný vykresliť ľubovoľný počet snímkov do 24 hodín spustením viacerých úloh.

Ďalší krok bolo vytvoriť dostatočne veľkú tréningovú množinu, ktorá bola zvolená nasledovne:

1. Reálne snímky

1 000 reálnych snímkov zozbieraných robotom.

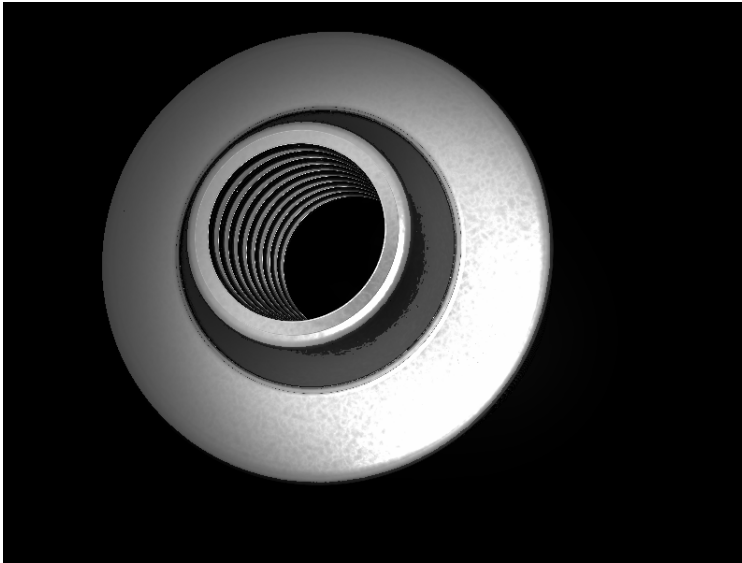
2. Umelé snímky podľa reálnych

1 000 umelých snímkov, ktoré odpovedajú reálnym snímkom (1).

3. Umelé snímky

10 000 úplne syntetických snímkov, ktoré simulujú polohu kamery voči objektu vo vzdialenosti od 25 do 35 cm od objektu. Rozpätie náklonu súčiastky je od 0 do 30 stupňov.

Na vytvorenie dát bol využitý python skript, ktorý využíva yaml súbory príslušiace reálnym snímkom (1.) a prevádza súradnicovú sústavu využívanú robotom do súradníc používaných programom POV-Ray [real_to_pov.py]. Zároveň generuje povray súbory, ktoré modelujú reálne snímky (2.). Ďalší python skript [position_generator.py] slúži na vygenerovanie umelých dát (3.) so systematickou voľbou polohy kamery a orientácie objektu.



Obrázok 3.4: Vytvorený model

■ 3.1.1 Detekcia polohy a orientácie objektu

K detekcii polohy daného objektu bola použitá neurónová sieť, ktorá je detailne popísaná v diplomovej práci [15]. Tá k určeniu potrebných informácií využíva 3 hlavné časti:

■ Detekcia objektu

Cieľom tejto časti je spracovať surový snímok z kamery a nájsť objekty na snímku. Ich pozícia je daná obdĺžnikovou oblasťou na snímke, nazývanou "bounding box", teda ohraničená oblasť. K tomu sa využíva detektor využívajúci histogram orientovaných gradientov (HOG). Detektor ale ešte neslúži k určeniu polohy snímaného objektu, ale skôr k určeniu, kde na snímku sa objekt nachádza. Výstupné snímky tohto detektoru v rozlíšení 64x64 pixelov sa používajú v ďalšom kroku procesu.



Kapitola 4

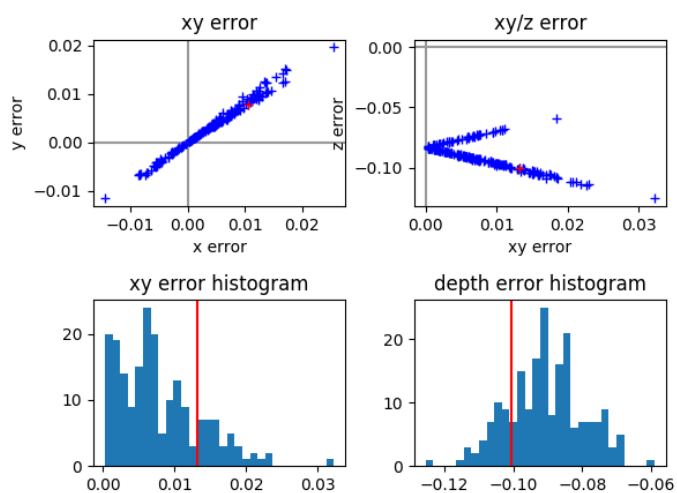
Praktická časť



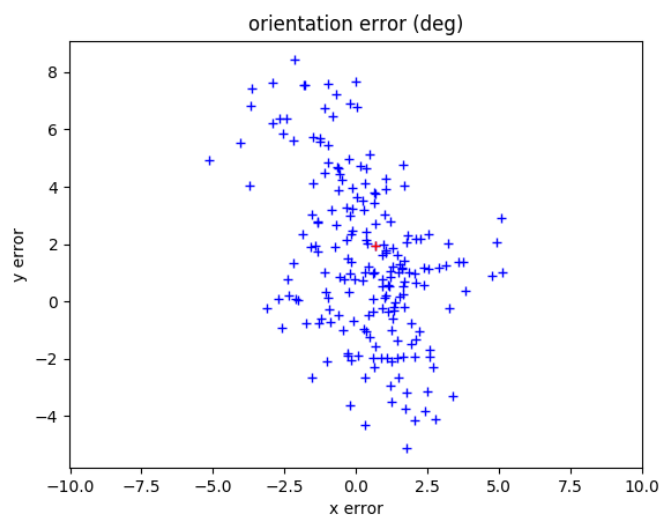
4.1 Experimenty a výsledky

Ako experimenty boli zvolené porovnanie výsledkov učenia na reálnych a umelých dátach. Všetky klasifikátory boli testované na rovnakých 200 reálnych snímkoch, ktoré neboli použité ako tréningová množina pre neurónovú sieť v žiadnom z učení. Na začiatku sú uvedené výsledky učenia neurónovej siete na reálnych dátach 4.1, 4.2, 4.3, 4.4.

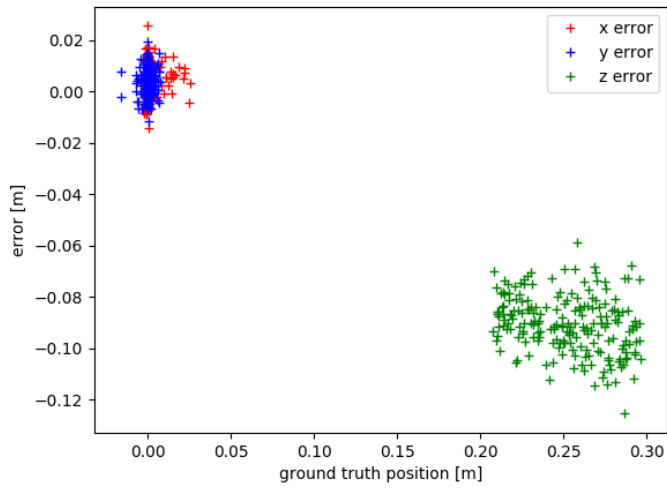
Všetky učenia prebehli na vyššie uvedenej neurónovej sieti s počtom iterácií 5 000 000, dropout konštanta má hodnotu 0.5 a učiacu konštantu 0.001.



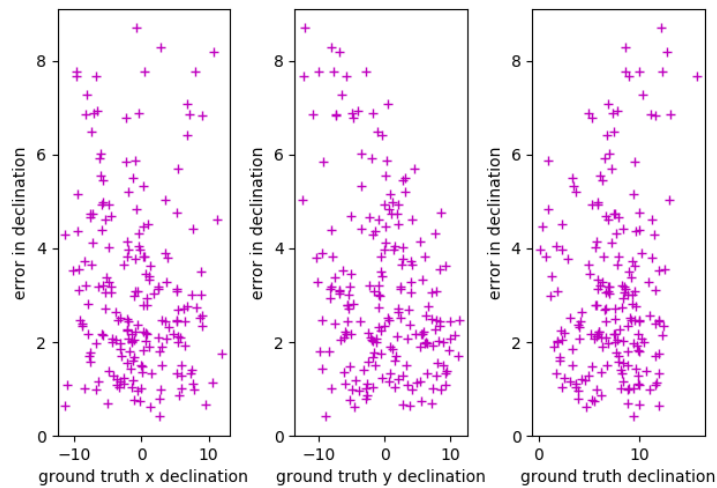
Obrázok 4.1: Graf výsledku učenia na reálnych dátach č. 1



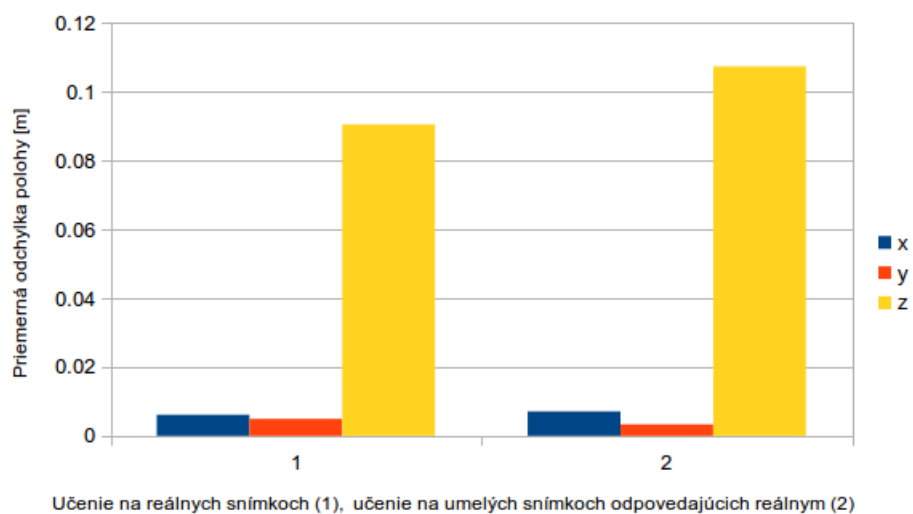
Obrázok 4.2: Graf výsledku učenia na reálnych dátach č. 2



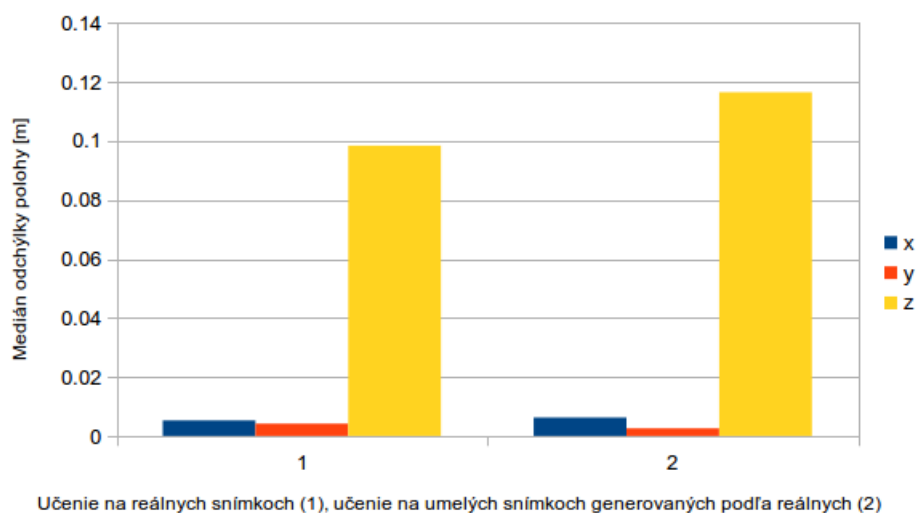
Obrázok 4.3: Graf výsledku učenia na reálnych dátach č. 3



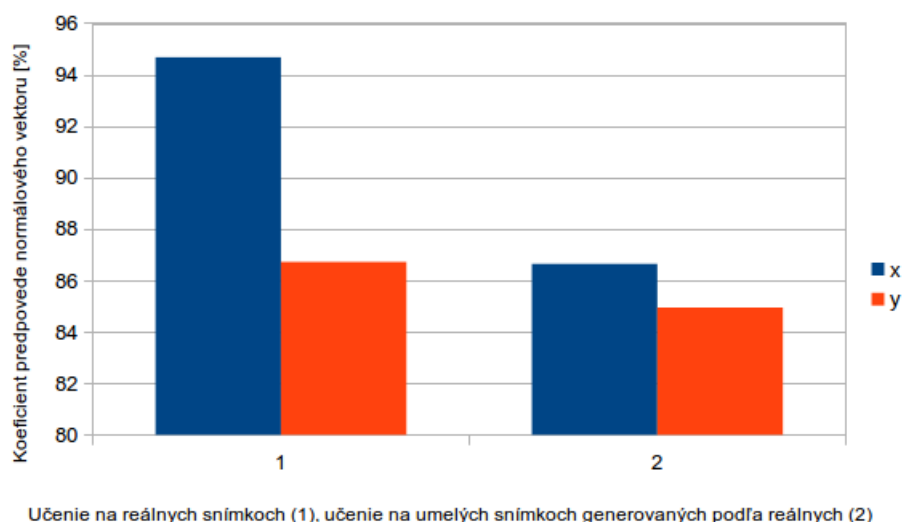
Obrázok 4.4: Graf výsledku učenia na reálnych dátach č. 4



Obrázok 4.5: Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych



Obrázok 4.6: Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych



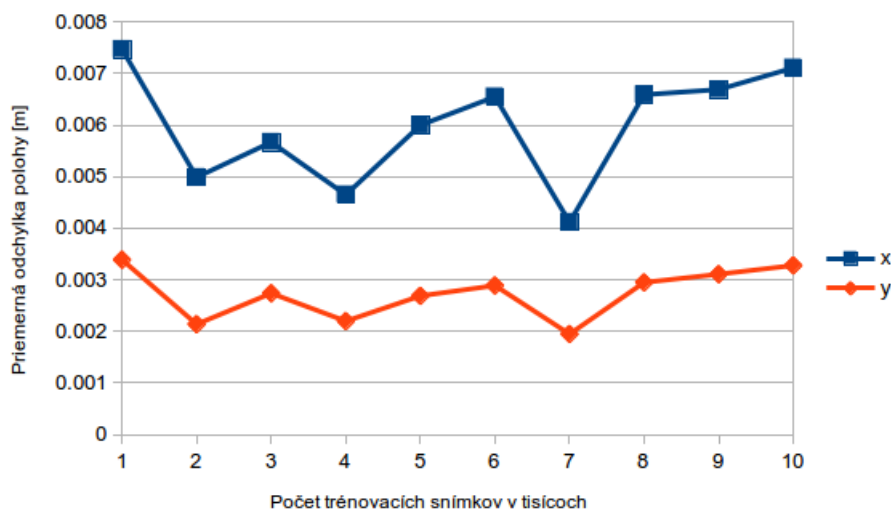
Obrázok 4.7: Porovnanie klasifikátorov naučených na reálnych a umelých dátach vygenerovaných podľa reálnych

Výsledky testovania klasifikátoru naučenom na reálnych dátach slúžia ako základ, z ktorého som vychádzal. Dôležité sú hodnoty predpovede zložiek normálového vektoru objektu, teda jeho poloha voči kamere, čo je v tomto prípade u x zložky 94.7% a u y zložky 86.7%. Zložka z je v tomto prípade nepodstatná, keďže sa jedná o objekt symetrický. Použitá sieť namiesto toho predpovedá koeficient použitý k výpočtu vzdialenosti objektu od kamery. Z grafu vidíme, že vzdialenosť nie je určená presne, čo je, ale známy problém použitej siete. Priemerná odchýlka x zložky vektoru polohy je 0.00614 metra, u y je 0.00495 metra a u z zložky 0.09048 metra. Mediány odchýliek vektoru polohy sú pre x 0.00525 metra, y 0.00495 metra a u z 0.09836 metra. Na grafoch 4.5, 4.6, 4.7 je porovnanie učenie na reálnych dátach s učením na dátach umelých, vygenerovaných podľa reálnych. Toto porovnanie ukazuje približne akú stratu presnosti prinášajú umelé dáta voči reálnym.

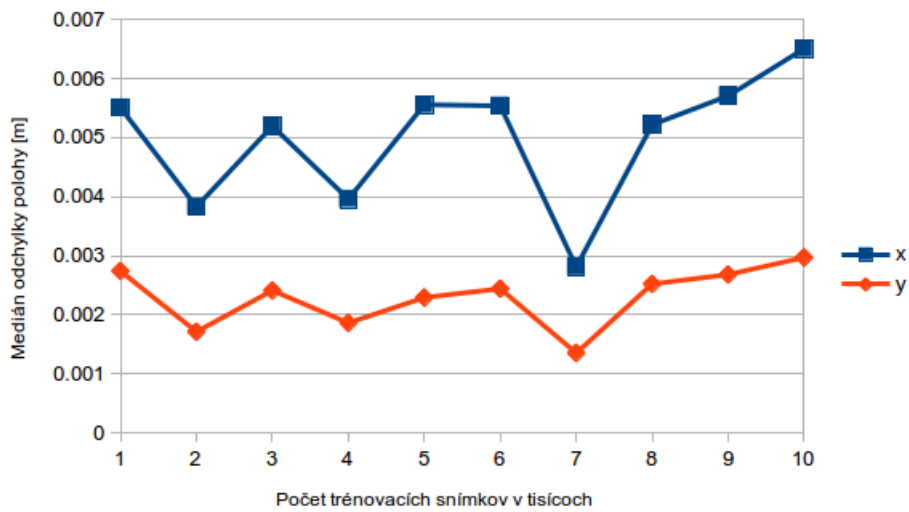
Výsledky testovania klasifikátoru naučeného na čisto umelých dátach je vidieť na grafoch 4.8, 4.9, 4.10, 4.11. Pri nižších počtoch umelých dát dochádza k veľkým rozdielom napriek rovnomernej voľbe tréningových dát z generovanej množiny. Po piatich tisícoch snímkov už dochádza k ustáleniu presnosti klasifikátorov s odchýlkou pri siedmich tisícoch snímkov. Aj napriek viacerým pokusom sa pri 7 tisíc snímkoch presnosť nevylepšila. Najlepšie

výsledky boli dosiahnuté pri 3 tisícoch snímok, kde koeficienty predpovede normálového vektoru orientácie objektu u zložky x sú na hodnote 88.2% a u y 83.5%. Zníženie presnosti oproti klasifikátoru naučenom na reálnych snímkoch pripisujem chýbajúcej z zložke normálového vektora, ktorá bola pri umelých snímkoch vždy nulová. Pričom pri reálnych snímkoch sa pohybovala v desatinách radiánu. Rotácia podľa osi z je totiž redundantná, keďže sa jedná o objekt symetrický, no uchycovač robota sa v reálnom svete otáča aj v tejto ose, aj keď to nie je nutné.

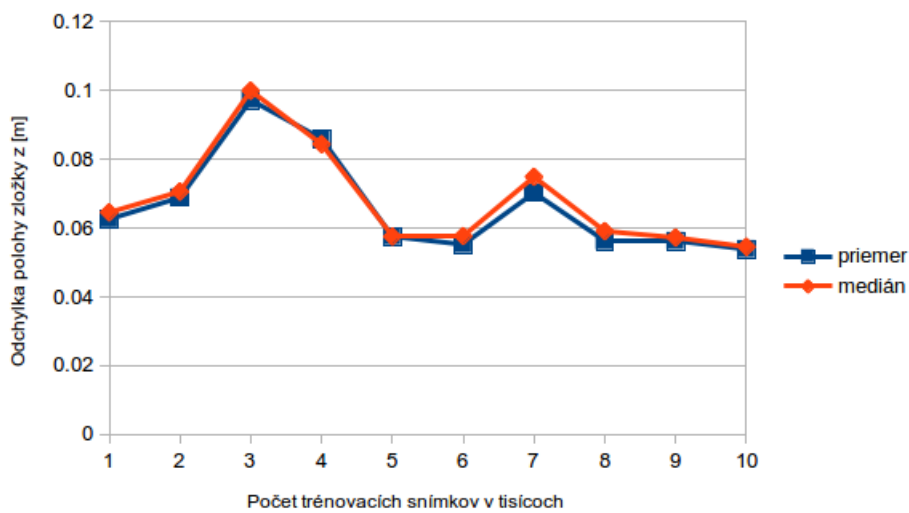
Posledným experimentom je kombinácia učenia neurónovej siete na umelých aj reálnych snímkoch. Na učenie boli použité všetky reálne snímky a postupne k nim boli pridávané umelé snímky po 1 000 kusov. Výsledky sú na grafoch 4.12, 4.13, 4.14, 4.15. Najlepšie výsledky boli dosiahnuté pri 2. pokuse, teda 2 000 umelých a 1 000 reálnych snímkov, a to koeficient predpovede normálového vektoru objektu dosahoval hodnôt pre x zložku 94.9% a u y 89.5%.



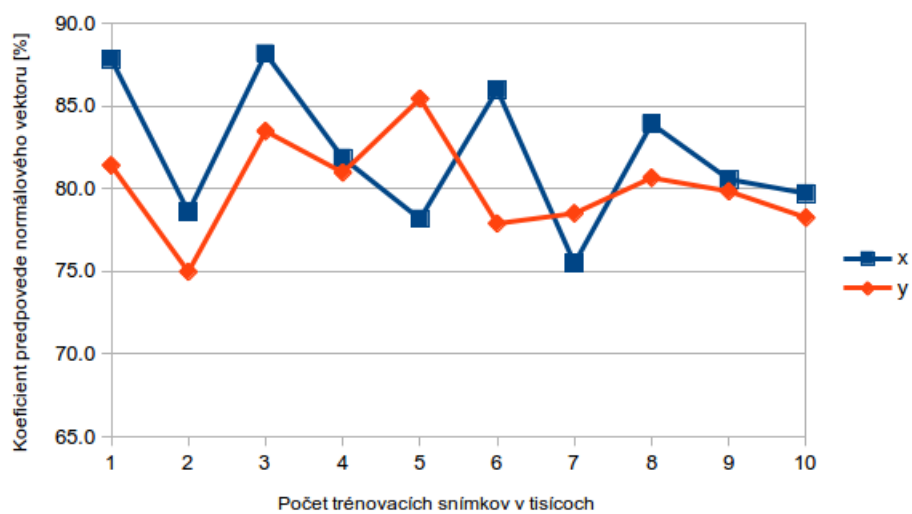
Obrázok 4.8: Priemerné odchylky x a y zložiek polohy objektu pri učení na umelých snímkoch



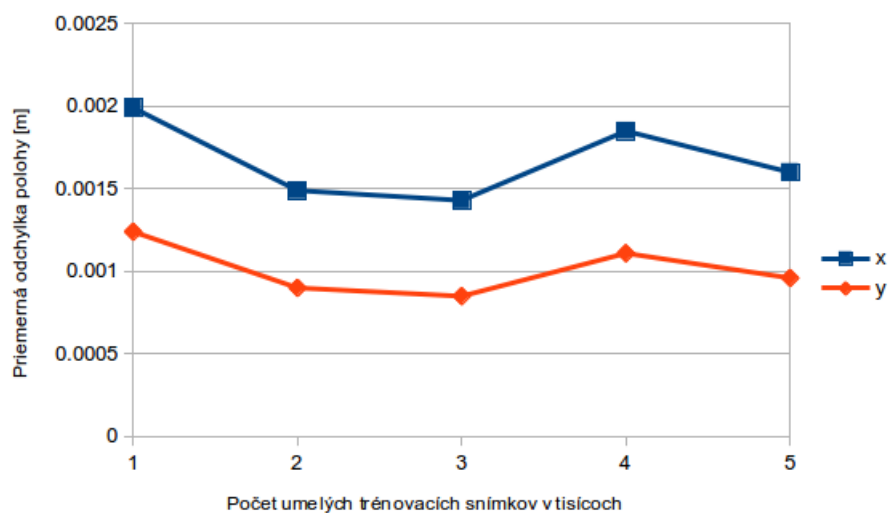
Obrázok 4.9: Medián odchyliek x a y zložiek polohy pri učení na umelých snímkoch



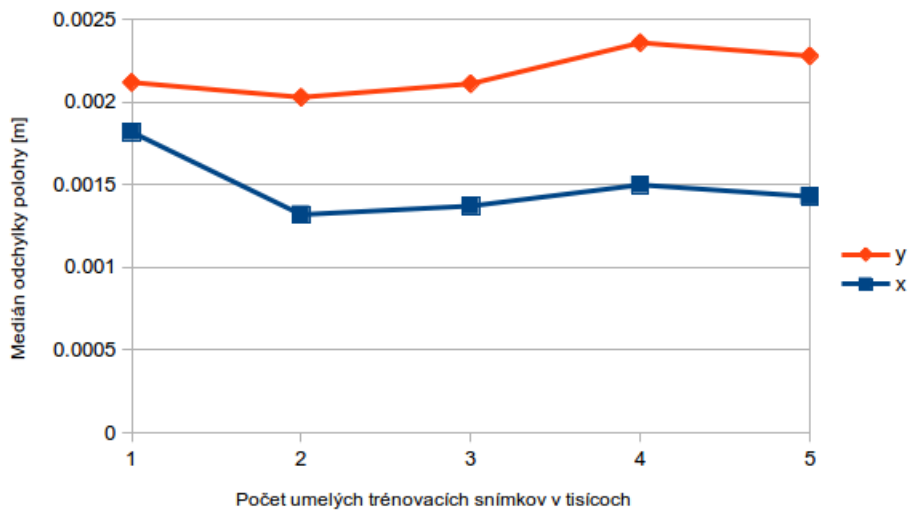
Obrázok 4.10: Priemerné odchylky a ich medián z zložky polohy pri učení na umelých snímkoch



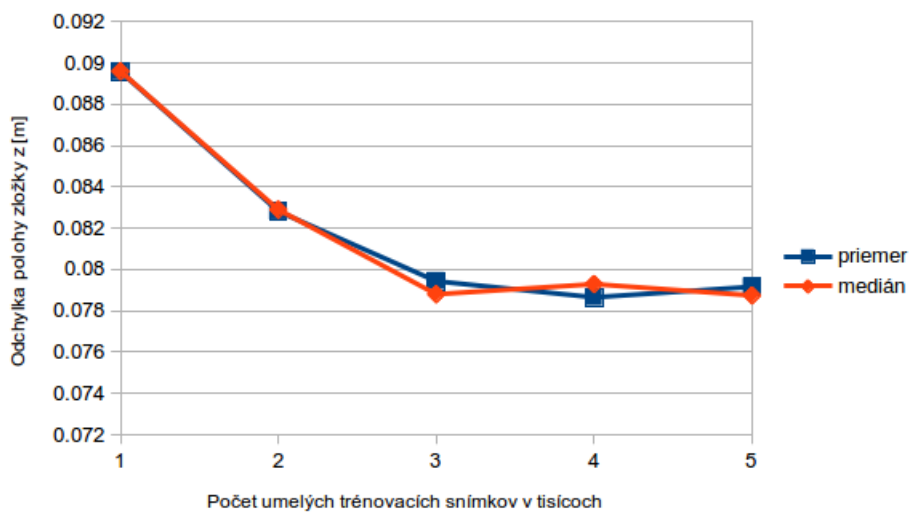
Obrázok 4.11: Koeficienty predpovede x a y zložiek polohy normálového vektoru pri učení na umelých snímkoch



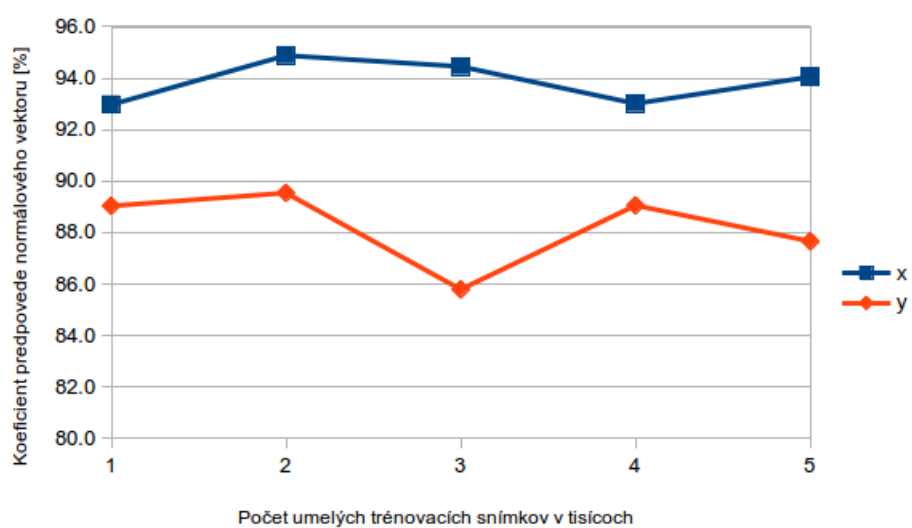
Obrázok 4.12: Priemerné odchylky x a y zložiek polohy pri učení na umelých a reálnych snímkoch



Obrázok 4.13: Medián odchýliek x a y zložiek polohy pri učení na umelých a reálnych snímkoch

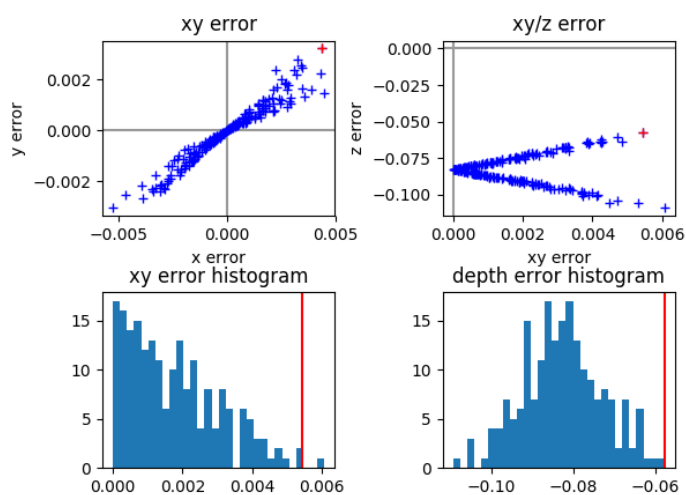


Obrázok 4.14: Priemerné odchýlky a ich medián z zložky polohy pri učení na umelých a reálnych snímkoch

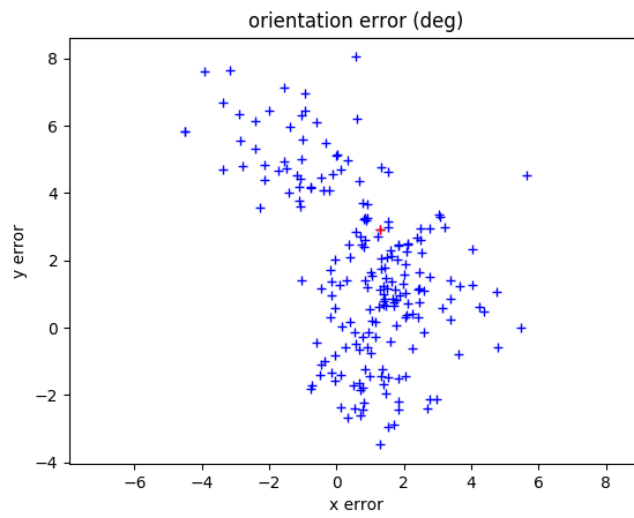


Obrázok 4.15: Koeficienty predpovede x a y zložiek polohy normálového vektora pri učení na umelých a reálnych snímkoch

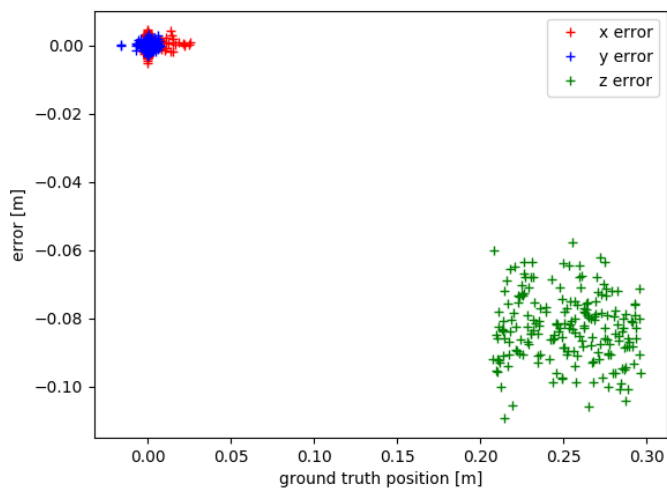
Na grafoch 4.16, 4.17, 4.18, 4.19 sú výsledky najlepšieho klasifikátora, teda použitie 1 000 reálnych snímkov a 2 000 umelých.



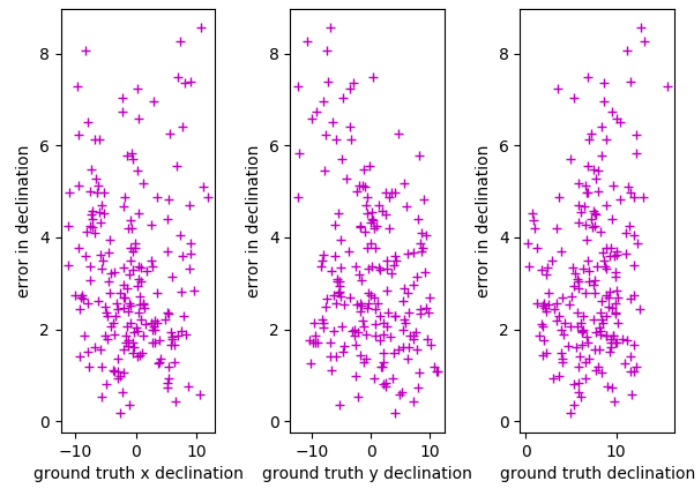
Obrázok 4.16: Graf výsledku učenia na kombinácii reálnych a umelých dát č. 1



Obrázok 4.17: Graf výsledku učenia na kombinácii reálnych a umelých dát č. 2



Obrázok 4.18: Graf výsledku učenia na kombinácii reálnych a umelých dát č. 3



Obrázok 4.19: Graf výsledku učenia na kombinácii reálnych a umelých dát č. 4



Kapitola 5

Záver

Cieľom tejto bakalárskej práce bolo zoznámiť sa s využitím simulovaných dát pri učení neurónových sietí. Ďalej pre prípad určovania polohy a orientácie objektu pre jeho následné uchopenie a zodvihnutie robotickým ramenom, vytvoril umelú trénovaciu množinu a porovnať výsledky oproti reálnym dátam. Túto úlohu sa mi podarilo splniť. Našiel som spôsob efektívnej tvorby veľkého množstva umelých dát s úplnou kontrolou nad prostredím. Vytvoril som množinu 10 000 umelých snímok, ktorú som využil na učenie neurónovej siete a klasifikátor otestoval na reálnych snímkoch. Oproti reálnym trénovacím dátam došlo k úbytku presnosti, a to v priemere o 9.5%. V najlepšom prípade len o 4%.

Pri kombinácii umelých a reálnych dát ako trénovacej množiny pre neurónovú sieť došlo k zlepšeniu presnosti v priemere o 0.5%. V najlepšom prípade o 1.5%.

V budúcnosti by sa mohol tento systém testovať pri meniacich sa svetelných podmienkach, na iných objektoch a prípadne rozšíriť aj o detekciu prekrývania objektov.



Literatúra

- [1] STERGIOU Christos a SIGANOS Dimitrios. Neural Networks. [cit. 2019-1-02] Dostupné z: https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol14/cs11/report.html
- [2] PINTO, Lerrel a Abhinav GUPTA. Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours. In: 2016 IEEE International Conference on Robotics and Automation (ICRA) [online]. IEEE, 2016, 2016, s. 3406-3413 [cit. 2019-01-02]. DOI: 10.1109/ICRA.2016.7487517. ISBN 978-1-4673-8026-3. Dostupné z: <http://ieeexplore.ieee.org/document/7487517/>
- [3] ZENG, Andy, Kuan-Ting YU, Shuran SONG, Daniel SUO, Ed WALKER, Alberto RODRIGUEZ a Jianxiong XIAO. Multi-view self-supervised deep learning for 6D pose estimation in the Amazon Picking Challenge. In: 2017 IEEE International Conference on Robotics and Automation (ICRA) [online]. IEEE, 2017, 2017, s. 1386-1383 [cit. 2019-01-02]. DOI: 10.1109/ICRA.2017.7989165. ISBN 978-1-5090-4633-1. Dostupné z: <http://ieeexplore.ieee.org/document/7989165/>
- [4] VARLEY, Jacob, Jonathan WEISZ, Jared WEISS a Peter ALLEN. Generating multi-fingered robotic grasps via deep learning. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Sys-

- tems (IROS) [online]. IEEE, 2015, 2015, s. 4415-4420 [cit. 2019-01-02]. DOI: 10.1109/IROS.2015.7354004. ISBN 978-1-4799-9994-1. Dostupné z: <http://ieeexplore.ieee.org/document/7354004/>
- [5] GUPTA, Saurabh, Pablo ARBELAEZ, Ross GIRSHICK a Jitendra MALIK. Aligning 3D models to RGB-D images of cluttered scenes. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [online]. IEEE, 2015, 2015, s. 4731-4740 [cit. 2019-01-02]. DOI: 10.1109/CVPR.2015.7299105. ISBN 978-1-4673-6964-0. Dostupné z: <http://ieeexplore.ieee.org/document/7299105/>
- [6] GUALTIERI, Marcus, Andreas TEN PAS, Kate SAENKO a Robert PLATT. High precision grasp pose detection in dense clutter. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) [online]. IEEE, 2016, 2016, s. 598-605 [cit. 2019-01-02]. DOI: 10.1109/IROS.2016.7759114. ISBN 978-1-5090-3762-9. Dostupné z: <http://ieeexplore.ieee.org/document/7759114/>
- [7] EITEL, Andreas, Jost Tobias SPRINGENBERG, Luciano SPINELLO, Martin RIEDMILLER a Wolfram BURGARD. Multimodal deep learning for robust RGB-D object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) [online]. IEEE, 2015, 2015, s. 681-687 [cit. 2019-01-02]. DOI: 10.1109/IROS.2015.7353446. ISBN 978-1-4799-9994-1. Dostupné z: <http://ieeexplore.ieee.org/document/7353446/>
- [8] DALAL, N. a B. TRIGGS. Histograms of Oriented Gradients for Human Detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) [online]. IEEE, 2005, s. 886-893 [cit. 2019-01-02]. DOI: 10.1109/CVPR.2005.177. ISBN 0-7695-2372-2. Dostupné z: <http://ieeexplore.ieee.org/document/1467360/>
- [9] BOLLES, Robert C. a Patrice HORAUD. 3DPO: A Three-Dimensional Part Orientation System. KANADE, Takeo, ed. Three-Dimensional Machine Vision [online]. Boston, MA: Springer US, 1987, 1987, s. 399-450 [cit. 2019-01-02]. The Kluwer International Series in Engineering and Computer Science. DOI: 10.1007/978-1-4613-1981-8_10. ISBN 978-1-4612-9184-8. Dostupné z: http://link.springer.com/10.1007/978-1-4613-1981-8_10

- [10] RAHARDJA, K. a A. KOSAKA. Vision-based bin-picking: recognition and localization of multiple complex objects using simple visual cues. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96 [online]. IEEE, 1996, s. 1448-1457 [cit. 2019-01-02]. DOI: 10.1109/IROS.1996.569005. ISBN 0-7803-3213-X. Dostupné z: <http://ieeexplore.ieee.org/document/569005/>
- [11] LIU, Ming-Yu, Oncel TUZEL, Ashok VEERARAGHAVAN, Yuichi TAGUCHI, Tim K MARKS a Rama CHELLAPPA. Fast object localization and pose estimation in heavy clutter for robotic bin picking. The International Journal of Robotics Research [online]. 2012, 31(8), 951-973 [cit. 2019-01-02]. DOI: 10.1177/0278364911436018. ISSN 0278-3649. Dostupné z: <http://journals.sagepub.com/doi/10.1177/0278364911436018>
- [12] PINTO, Nicolas, Youssef BARHOMI, David D. COX a James J. DICARLO. Comparing state-of-the-art visual features on invariant object recognition tasks. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV) [online]. IEEE, 2011, 2011, s. 463-470 [cit. 2019-01-02]. DOI: 10.1109/WACV.2011.5711540. ISBN 978-1-4244-9496-5. Dostupné z: <http://ieeexplore.ieee.org/document/5711540/>
- [13] MIT Technology review, Deep Learning With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart. [cit. 2019-1-02], Dostupné z: <https://www.technologyreview.com/s/513696/deep-learning/>
- [14] Michael Nielsen: How the backpropagation algorithm works, [cit. 2019-1-02], Dostupné z: <http://neuralnetworksanddeeplearning.com/chap2.html>
- [15] Sushkov, R.: Detection and Pose Determination of a Part for Bin Picking. Ph.D. thesis, Czech Technical University in Prague (6 2017), <https://dspace.cvut.cz/handle/10467/68468?show=full>
- [16] Hynek, D.: Tracking of flying drones. Bachelor thesis, Czech Technical University in Prague (5 2018), <http://hdl.handle.net/10467/76644>

- [17] Miroslav Surák, Karel Košnar, Miroslav Kulich, and Libor Přeučil: Visual Data Simulation for Deep Learning in Robot Manipulation Tasks, Faculty of Electrical Engineering Czech Technical University in Prague, Czech Institute of Informatics, Robotics and Cybernetics Czech Technical University in Prague, <http://imr.ciirc.cvut.cz>
- [18] Persistence of Vision Pty. Ltd.: POV-Ray - The Persistence of Vision Raytracer (2004), <http://www.povray.org/>
- [19] John Skinner, Sourav Garg, Niko Sunderhauf, Peter Corke, Ben Upcroft, Michael Milford: High-Fidelity Simulation for Evaluating Robotic Vision Performance, 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon Convention Center, October 9-14, 2016, Daejeon, Korea [cit. 2019-1-02]
- [20] Xingchao Peng, Baochen Sun, Karim Ali, Kate Saenko: Learning Deep Object Detectors from 3D Models, University of Massachusetts Lowell [cit. 2019-1-02]
- [21] Max Jaderberg, Karen Simonyan, Andrea Vedaldi. Andrew Zisserman: Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition, Visual Geometry Group, University of Oxford
- [22] CS231n Convolutional Neural Networks for Visual Recognition [cit. 2019-1-02] Dostupné z: <http://cs231n.github.io/neural-networks-1/>
- [23] MetaCentrum CESNET (CESNET LM2015042) [cit. 2019-1-02], Dostupné z: <https://www.metacentrum.cz/cs/index.html>
- [24] Wikipedia, Perceptron, [cit. 2019-1-02], Dostupné z: <https://en.wikipedia.org/wiki/Perceptron>



Dodatok A

Obsah priloženého CD

Miroslav Surák

- └─ BP_Miroslav_Surák.pdf
- └─ výsledky učenia
 - └─ výsledky učenia na kombinácií reálnych a umelých snímkov
 - └─ výsledky učenia na umelých snímkoch
- └─ skripty
 - └─ position_generator.py
 - └─ real_to_pov.py
 - └─ job.sh
 - └─ disc.pov