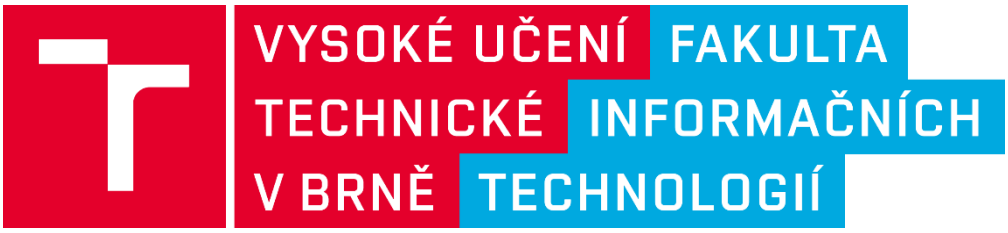


Technická zpráva k projektu
VI20172020068

Nástroje a metody zpracování videa a obrazu pro zvýšení efektivity operací bezpečnostních a záchranných složek (VRASSEO)

Generátor a detektor modelů zbraní

Leden 2019

Abstrakt

Tato zpráva obsahuje popis metod pro detekci, klasifikaci typu (délky) a orientace (natočení) zbraně. Jsou popsány jednotlivé typy zbraní, metody strojového učení jak klasických, tak využívajících neuronové sítě. U neuronových sítí jsou pak popsány typy jednotlivých vrstev i používané architektury. Popsány jsou i detektory objektů. Pro každou část (detekce, typ i orientace) je navrženo několik přístupů. Ty jsou otestovány na volně dostupných datech. Pro natočení je dataset prakticky vytvořen za pomocí automatického generování. Na závěr jsou diskutovány výsledky, u detekce 97,8 %, u délky 84,14 % a u natočení 75,14 %.

Obsah

1 Úvod ... 3

2 Klasifikace zbraní ... 3

3 Teoretický rozbor technické zprávy ... 4

3.1 Detekce zbraní ... 4

3.1.1 Klasické metody strojového učení ... 4

3.1.2 Konvoluční neuronové sítě ... 6

3.2 Detektory objektů ... 7

3.2.1 Sliding window ... 7

3.2.2 Region proposals ... 8

3.2.3 YOLO ... 8

3.2.4 SSD ... 9

4 Návrh řešení ... 9

4.1 Detekce zbraní ... 10

4.1.1 Klasické metody (HOG a SVM) ... 10

4.1.2 YOLO ... 11

4.1.3 Model s využitím CNN a sliding window ... 12

4.2 Popis zbraní ... 13

4.2.1 Typ (délka) zbraně ... 15

4.2.2 Orientace ... 15

5 Vyhodnocení modelů ... 16

5.1 Detekce zbraní ... 16

5.1.1 HOG a SVM ... 16

5.1.2 Yolov3 ... 17

5.1.3 CNN VGG16 ... 17

5.2 Popis zbraní ... 18

6 Závěr ... 20

1 Úvod

Detekce a prevence hrozeb, představuje čím dál významnější část povinností bezpečnostních složek. Díky neustále se zvyšujícímu množství videokamer a fotoaparátů, ať již bezpečnostních CCTV (close circuit television) kamer, či komerčních senzorů v mobilních telefonech a jejich stoupající kvalitě, stoupá zároveň poptávka po automatickém zpracování těchto dat, pro bezpečnostní účely.

Ať se jedná o real-time analýzu streamovaného videa či o zpětnou analýzu velkého objemu obrazových záznamů, schopnost rychle a kvalitně v materiálech odhalit palné zbraně a získat informace které pomohu s vyhodnocením hrozby představuje silný a žádaný nástroj.

Tato zpráva představuje popis problematiky a rozbor metod řešení využívající jak klasické metody strojového učení, tak postupy založené na konvolučních neuronových sítích (CNN).

V této zprávě dále představujeme nástroje, které byly pro zmíněné úlohy připraveny. Jedná se o vývoj nástrojů pro detekci zbraní v obraze. Nástroj pro určení typu zbraně a určení orientaci zbraně v obraze. Příprava obrazových dat pro algoritmy strojového učení byla realizována pomocí obrazového generátoru SYDAgenerátor.

Dosažené výsledky a diskuze nad dalším směřování vývoje je taktéž součástí této zprávy.

2 Klasifikace zbraní

Definice zbraně dle § 118, zákona č. 40/2009 zní: „zbraní se tu rozumí, pokud z jednotlivého ustanovení trestního zákona nevyplývá něco jiného, cokoli, čím je možno učinit útok proti tělu důraznějším.“ V této zprávě se zabýváme primárně zbraněmi střelnými. Kde zbraň považujeme za střelnou, pakliže vymršťuje na dálku střelu rozrušující svou dopadovou energií zasažený cíl. Důležité je také zmínit pojem palná zbraň – střelná zbraň, kde vymrštění probíhá okamžitým uvolněním chemické energie. Jako střelná zbraň je totiž klasifikován i například luk.

Dále se v této práci zaměřujeme na ruční zbraně, které na rozdíl od lafetovaných uzpůsobené tak, aby je mohla ovládat a přenášet jedna osoba. Následující dělení pak představuje popis, kterým se v práci zabýváme.

Klasifikace dle délky:

- Krátké zbraně: Palné zbraně, jejichž délka hlavně nepřesahuje 300 mm nebo jejichž celková délka nepřesahuje 600 mm.
- Dlouhé zbraně: Palné zbraně, které nejsou krátkými zbraněmi

Klasifikace zbraní dle charakteru střelby:

- Jednoranové zbraně: palné zbraně bez zásobníku nebo jiného podávacího ústrojí, u nichž se opětovné nabití děje ručním vložením náboje do nábojové komory, hlavně nebo nábojiště.
- Víceranné zbraně: palné zbraně bez zásobníku nebo jiného podávacího ústrojí, s 2 nebo více hlavněmi, u níž se opětovné nabití děje ručním vložením nábojů do nábojových komor, hlavní nebo nábojišť.
- Opakovací zbraně: palné zbraně se zásobníkem nebo jiným podávacím ústrojím, u níž se opětovné nabití děje v důsledku ručního ovládání závěru nebo mechanického otočení revolverového válce.
- Samonabíjecí zbraně: palné zbraně, u nichž se opětovné nabití děje v důsledku předchozího výstřelu a u kterých konstrukce neumožňuje více výstřelů na jedno stisknutí spouště.
- Samočinné zbraně: palné zbraně, u nichž se opětovné nabití děje v důsledku předchozího výstřelu a u kterých konstrukce umožňuje více výstřelů na jedno stisknutí spouště. Dlouhé zbraně: Palné zbraně, které nejsou krátkými zbraněmi.

Norma ČSN 39 5002-1 slučuje kategorie „samonabíjecí“ a „samočinná“ do jedné s označované jako „automatická“.

3 Teoretický rozbor technické zprávy

V dnešním světě, je denně potřeba realizovat miliony detekcí, a to už není možné bez vhodných přístrojů a zobrazovacích metod, proto zde popsané metody, eliminují potřebu aktivního operátora pro hrubá data, a umisťují ho až do rozhodovací pozice kde je jeho úkolem vyhodnotit předložené informace. Protože detekce zbraní a jejich následný popis pomocí počítačového vidění je poměrně rozsáhlé téma, v následující části zpráva obsahuje popis v současnosti využívaných postupů, a state-of-the-art postupy které umožnují zpracovávání většího množství dat.

3.1 Detekce zbraní

V této zprávě se soustředíme na metody využívající data ze senzorů pracujících ve viditelné časti spektra. Z pohledu použitých nástrojů je dělíme na klasické metody, tzn. bez použití neuronových sítí (angl. neural network – NN) a metody založené na využití neuronových sítí. Ty se ještě dále člení podle použité architektury. Nejprve uvedeme příklady klasických metod, v další části řešení s použitím neuronových sítí.

3.1.1 Klasické metody strojového učení

Klasickými metodami nazýváme řešení, které nevyužívají nyní velmi rozšířené neuronové sítě. Jedná se o metody, které jsou založeny na matematické analýzy s předvídatelným charakterem.

Detekce s využitím deskriptorů a klasifikátorů

Autoři [1] [2] vytvořili dva velmi podobné, až přímo totožné systémy, které se liší pouze v použití rozdílných příznakových detektorů a deskriptorů. [1] založil svou práci na příznakových deskriptoru a detektoru SURF. V [2] byl použit Fast Retina Keypoint deskriptor (FREAK) a Harris interest point detektor.

Systém si na začátku načte deskriptor zbraní, následně předzpracovat obrázky, tj. odstraní šum a upraví velikost na 400x300 pixelů. Na obrázcích je následně aplikována segmentace na základě barvy za použití k-means shlukování. Na výstup tak systém dostává shluky barev, které jsou podobné zbraním (v tomto případě černá). Protože v důsledku šumu mohly vzniknout malé oblasti dané barvy, pokračuje extrahovaného spojitých oblastí, jejichž obsah je větší než 1 000 pixelů, které tímto krokem odseparují. Nastává morfologické uzavření vybraných oblastí, z něhož detektor extrahován příznaky tvaru objektu, které porovná s již načteným deskriptory a pokud se shoduje alespoň 50 % příznaků, systém detekuje objekt jako zbraň.

Systémy dosahují velmi podobné celkové úspěšnosti, a to 84.26 % a 88.67 %. Výhody těchto řešení jsou invariantnost vůči velikosti a rotaci, a také možnost detekovat několik objektů v 1 obraze. Protože systémy nejprve aplikují segmentaci na základě barvy, jsou proto výrazně závislé na barvě zbraní a nejsou schopné detekovat zbraně atypické barvy.

Alternativním přístupem je pak využití deskriptoru Histogram of Oriented Gradients (HOG), který je vhodný díky své výpočetní nenáročnosti, avšak je zde třeba vzít v potaz rotační ne-invariantnost.

Support Vector Machine (SVM) představuje metodu lineární klasifikace. Cílem je najít nadrovinu parametrů, která prostor příznaků optimálně rozdělí tak, že trénovací data, které patří různým třídám budou ležet v opačných poloprostoru. Jinými slovy optimální nadrovina je taková, kde minimální vzdálenosti bodů od roviny jsou co největší. Na popis nadroviny stačí pouze body ležící na jejím okraji a těch je obvykle málo. Jmenují se podpůrné vektory (angl. Support Vectors).

Klasifikace zahrnuje trénování a testování dat. Data na trénování jsou určeny dvojicí (x_i, y_i) pro $i = 1,2, ..., n$ kde, $y_i \in \{-1,1\}$. x_i je vektor dimenze, který popisuje vlastnosti daného prvku. y_i určuje, do které množiny vektor patří. Testovací data tedy obsahují pouze vektory x, kde cílem klasifikátor je určit jeho příznak y.

Obvykle není možné rozdělit trénovací data lineárně. Proto se používá namapování dat (příznaků) do vyšší dimenze, kde se rozdělí nadrovinou. Takto se z lineárně nespravovatelné úlohy stává lineární oddělitelná. Pro zjednodušení a vyřešení klasifikačního problému se používají jaderné funkce. Výběr funkce je však závislý na povaze trénovacích dat, a častokrát je časově náročné najít nejvhodnější funkci.

Další klasifikační metodou je k-nearest-neighbour (k-nejbližších-sousedů, KNN). Je to metoda učení s učitelem. Klasifikace probíhá ve dvou krocích, v první se najde k nejbližších sousedů s trénovacích dat. V druhé se pomocí většinového hlasování se přiřadí očekávaná třída. Nízké k má často za následek nedostatečnou schopnost generalizace.

3.1.2 Konvoluční neuronové sítě

Konvoluční neuronové sítě představují v současnosti nejzkoumanější princip zpracovávání dat. Své využití nacházejí často při zpracování obrazu, vzhledem k návrhu, který je vhodný pro zpracování jedno a dvojrozměrných dat. Struktura CNN využívá tří typů vrstev.

Konvoluční vrstva
Sestává ze souboru filtrů, které jsou schopné se učit. Každý filtr je prostorově malý (podél výšky a šířky), ale prochází přes celou hloubku vstupního objemu (konkrétně u RGB obrazu přes všechny 3 kanály barev). Při posouvání filtru přes šířku a výšku vstupního objemu vytvoříme dvourozměrnou aktivační mapu, která definuje hodnotu tohoto filtru v každé prostorové poloze. Síť tedy naučí filtry, aby se aktivovaly, když vidí konkrétní typ vizuálně prvku jako například hranu nebo skvrnu barvy.

Výstupní objem neuronů dané vrstvy je závislý na vstupu a 3 hyperparametrech: hloubce (v tomto případě hloubka reprezentuje počet filtrů, které chceme použít, každý se učí hledat jiné příznaky na vstupu), kroku (určuje, o kolik pixelů se filtr posouvá v každém kroku přechodu) a nulovými okraji (angl. zero-padding). Pokud velikost vstupního objemu označíme W, velikost vnímavého pole neuronů F, krok S a nulové okraje P, velikost výstupního objemu určíme jako (1):

$$\frac{(W - F + 2P)}{S} + 1 \qquad (1)$$

Pooling vrstva
Se vkládá mezi jednotlivé konvoluční vrstvy. Úkolem pooling vrstvy je snižování prostorové velikosti a tím i počtu parametrů a výpočetní náročnosti. Také se používá jako opatření proti přeučení. V přítomnosti se na redukci převážně využívá metoda max. poolingu.

Plně-propojená vrstva (dense vrstva)
Vyskytuje se i v klasických NN. Vrstva, kde jsou neurony mezi dvěma sousedními vrstvami plně párově propojené, ale v rámci jedné vrstvy mezi sebou nemají žádné propojení [3].

Dropout vrstva
Jedná se vrstvu, která se používá ke snížení rizika přetrénování. Využívá se mezi plně-propojenými vrstvami, ale také po pooling vrstvě mezi jednotlivými konvoluční vrstvami.

Neznámější architektury:

Lene
Jde o první úspěšnou aplikaci neuronových sítí. Byla vyvinuta v devadesátých letech minulého století a používaná na rozpoznávání číslic.

AlexNet
První architektura, která výrazně zpopularizovala použití konvoluční neuronových sítí. Architektura je podobná Lene síti, akorát je větší, hlubší a úspěšně používá několik konvoluční vrstev za sebou, bez použití pooling vrstvy.

ZFNet
Jedná se o vylepšení AlexNetu, především rozšířením středních konvoluční vrstev a zmenšením rozměrů filtru a velikosti kroku v první vrstvě.

GoogLeNet
Hlavní přínos je ve vynalezení počátečního modulu, který výrazně redukuje počet parametrů v síti.

VGGNet
Ukazuje, že hloubka sítě je kritickým komponentem pro dosažení dobrých výsledků.

3.2 Detektory objektů

Detekce objektu v obraze sestává z rozpoznání objektu a jeho následného nalezení v obraze. Stávající metody řeší problém detekce jeho přepracování na klasifikační problém, kde se jako první krok natrénuje klasifikátor, který se v průběhu detekce spouští na několik oblasti vstupního obrazu pomocí zvoleného přístupu [4].

3.2.1 Sliding window

Jde o výpočetně náročnou metodu, která využívá velký počet oken (řádově až 10^4), kde by se principiálně mohl nacházet objekt, který se snažíme detekovat. Postupně skenuje vstupní obraz pomocí okna, které se posouvá a mění svou velikost. Nad každým oknem spouští klasifikátor, pomocí kterého se snažíme detekovat objekt. Příslušné práce věnující se této problematice zvyšují výkonnost pomocí vytváření sofistikovanějších klasifikátorů. Tento model dosahuje poměrně vysokou přesnost, avšak detekční proces je obvykle příliš pomalý na využití tohoto přístupu pro detekci objektů v reálném čase.

3.2.2 Region proposals

Namísto detekování objektu ve všech možných oknech tento přístup se zaměřuje na výběr kandidátních oken pomocí proposal detection metod. První model, který byl použit v neuronových sítích na základě tohoto přístupu byl Region-based CNN (R-CNN). Tento model vytváří kolem 2 000 potenciálních ohraničujících boxů, které vybere metodou selektivního vyhledávání. Vybrané oblasti zakreslí do obrazu, který následně předá výkonnému klasifikátor na bázi konvoluční neuronové sítě. Klasifikátor následně extrahuje příznaky a ohodnotí jednotlivé oblasti pomocí SVM, upraví ohraničující boxy pomocí lineárního modelu a eliminuje duplicitní detekce prostřednictvím non-max suppression. Dosahuje rychlosti detekce na dobře známé PASCAL-VOC přibližně 40 s na obrázek.

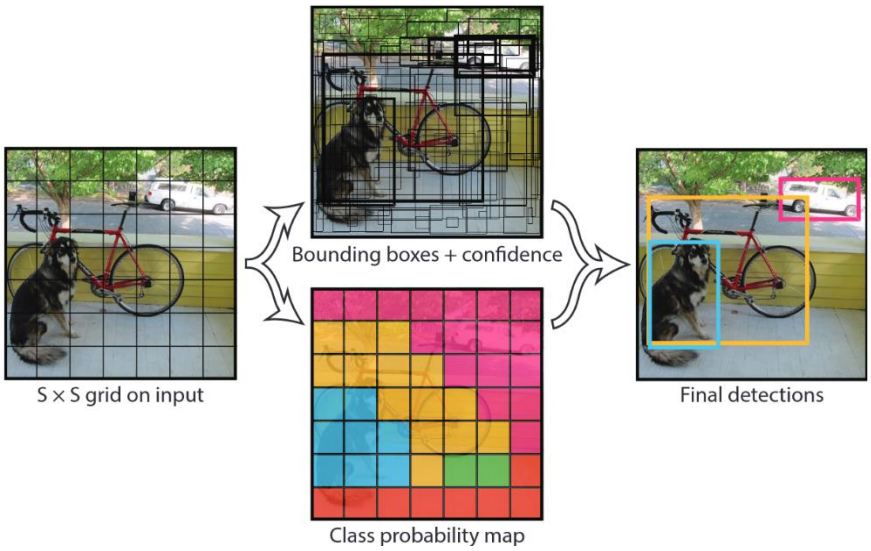
Tento model byl překonán vynálezem nového modelu Fast R-CNN a Faster R-CNN. Hlavním účelem těchto sítí je zrychlení detekce. Protože předchozí model vytvářel okolo 2 000 oblastí, docházelo k překrývání oblastí a s tím spojeným opakovaným výpočtem. Tento model to řeší pomocí techniky Region of Interest Pooling. Rovněž dochází ke zjednodušení modelu. R-CNN obsahuje zvláštní moduly pro extrakci příznaků, klasifikaci a ohraničení, Fast R-CNN to zvládá v rámci jedné sítě. Fast R-CNN dosahuje rychlosti 2 s na obrázek.

Nejnovějším modelem na bázi R-CNN je momentálně Faster R-CNN. Model nahrazuje selektivní vyhledávání. Je tvořen jednou hlubokou neuronových sítí. Mapa příznaků, která je tvořena CNN je použita na predikovaného oblastí. Dosáhlo se to přidáním plně-propojené vrstvy za konvoluční síť, čímž vzniká Region Proposal Network. Funguje na bázi posuvného okna, které pracuje nad mapou příznaků. Faster R-CNN zlepšuje výpočet, přístup k datům a využití disku realizované R-CNN. Faster R-CNN dosahuje rychlosti 140 ms na obrázek [4].

3.2.3 YOLO

You only look once (YOLO), je neuronová síť zaměřena na detekci objektů v reálném čase. YOLO představuje úplně jiný detekční přístup než předchozí. Na rozdíl on natrénovaní klasifikátoru, který je pak v krocích používaný pro detekci v různých částech obrazu, YOLO provádí celý akt v jednom kroku. Také zpracovává všechny objekty zároveň, a tak dosahuje vynikající rychlost při zachování relativně dobré přesnosti.

Vstupní obraz je rozdělen na mřížku o velikosti SxS. Jeden prvek této mřížky se nazývá buňka. Každá buňka rozděleného obrazu je zodpovědná za předpovídání pěti ohraničujících boxů. YOLO také vygeneruje skóre důvěry, které nám ukáže, jaké si je jisté, že předpovězené ohraničující boxy skutečně obklopují nějaký objekt. Toto skóre neříká nic o tom, jaký typ objektu je v boxu, vypovídá pouze o tom, zda je daný tvar boxu dobrý. Může to vypadat přibližně jako na obrázku 1.



Obrázek 1: Ukázka funkcionality YOLO (převzato z [1]).

Protože mřížka je o velikosti SxS máme dohromady S^2 buněk a každá buňka předpovídá pět ohraničujících boxů, tedy vyústí v 5 S^2 ohraničujících boxů. Ukazuje se, že většina těchto boxů bude mít velmi nízké skóre důvěry, takže pouze boxy, jejichž konečné skóre je vyšší než námi stanovený práh, budou považovány za detekované objekty. Pro každý ohraničující box buňka také určuje i třídu objektu. Dnes existuje řada variant YOLO sítí, jako například Yolov2, Yolov3, Tiny Yolo.

3.2.4 SSD

Jde o dopřednou NN. První vrstvy modelu jsou založeny na architektuře VGG16. Za nimi následuje podpůrná síť, která zajišťuje detekci. Na předchozí část je napojena konvoluční vrstvami, které postupně snižují velikost. Detekční model tak může predikovat nad několika vrstvami, čímž se odlišuje od YOLO sítě, která pracuje pouze s jednou vrstvou příznaků.

4 Návrh řešení

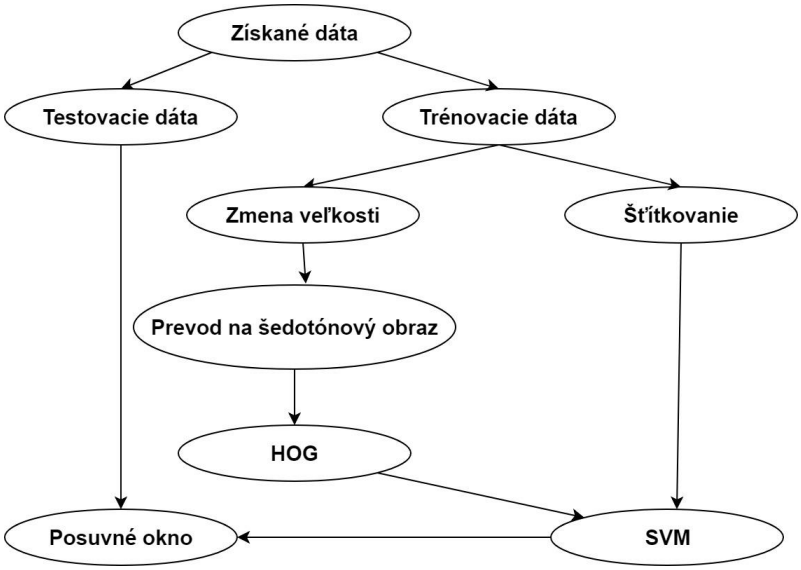
V této kapitole je popsán návrh metod, které řeší jak detekci zbraně, tak její délku a orientaci. Tato úloha je řešena ve dvou krocích. Detekce zbraní je složena, že tří odlišných modelů a metod. Jsou využity klasické metody a potom dva modely využívající konvoluční neuronové sítě. Jeden je zaměřen na přesnost detekce, zatímco druhý na real-time detekci. U každého modelu je uvedena jeho základní charakteristika, datasety, které byli využity a případné specifické informace. Modely pro popis zbraní jsou využívají dvě architektury neuronových sítí pro detekci i natočení zbraní. Při detekci se navíc využívá i klasického přístupu pomocí HOG a SVM resp. KNN.

4.1 Detekce zbraní

Prvním krokem je definice vhodného datasetu (tj. sada obrázků, které je využita na trénování, validaci a testování). Vzhledem k tomu, že jich volně k dispozici není mnoho (v dostatečném množství pro využití všemi metodami). Bylo rozhodnuto v první verzi využít detekci jen krátkých zbraní. Z pohledu statistik je tento typ zbraní nejčastěji používaný při páchání trestné činnosti. A tomu také odpovídá rozsah a množství dostupných datasetů.

4.1.1 Klasické metody (HOG a SVM)

Jako první je sestrojen model, který využívá klasických metod. Tj. metod, které nevyužívají konvoluční neuronové sítě. Mezi existujícími pracemi, které se zaobírají danou problematikou nebyla nalezena žádná, ve které použitá metoda na získání a popis příznaků využívala HOG transformace a zároveň žádnou studii, která by nevhodnost její aplikace zdůvodňovala. Z těchto důvodu byl využít právě tento deskriptor. Popis jednotlivých kroků vykonávaných v rámci prvního modelu je možné vidět na obrázku 2.



Obrázek 2: Sekvence kroků u modelu využívající klasické metody.

Pro tento model je využit dataset, který je dostupný z práce [4] (je připravený pro využití přístupu posuvného okna). Dohromady má 9 857 obrázku rozdělených do 102 tříd. Třída krátkých zbraní (AAAPistol) má 795 obrázků. Ostatní třídy obsahují obrázky, které nejsou zbraně. Pro tento model byl dataset upraven tak, že byl počet tříd zredukován na dvě (obsahuje zbraň a neobsahuje). Obrázky předzpracovány. první úpravou je transformace na stejnou velikost (128x128). Další změna je převedení do odstínů šedi.

Práce s modelem pokračuje tvorbou pole labelů pro vstupní data a pole deskriptorů pro všechny vstupní obrázků (k tomu se využívá HOG metody). Pro klasifikaci se využije SVM. K natrénování klasifikátoru se použije pole příznaků a labelů. Poslední důležitou částí je posuvné okno. Tam se místo obvyklé změny

velikosti okna upravuje velikost prohledávaného obrázku. Velikost okna tím zůstane fixní. K uložení a práci s různými velikostmi obrázků využíváme tzv. model obrazových pyramid. Je to víceúrovňová reprezentace obrázku. Ve spodní vrstvě se nachází původní velikost obrázku a v každé další zmenšenina toho původního, dokud se nedosáhne určená minimální velikost.

4.1.2 YOLO

Priorita tohoto modelu byla dosažení detekce zbraně v reálném čase. Pro otestování modelu takového typu je potřeba přihlédnout i časové náročnosti a limitech dostupného hardwaru. Po zvážení možností byla využita sít Yolov3, konkrétně její Tiny verze, která sice nedosahuje takovou přesnost, ale je poměrně malá. Díky tomu se dá v rozumném čase natrénovat. Architektura této sítě je zobrazena v tabulce 1.

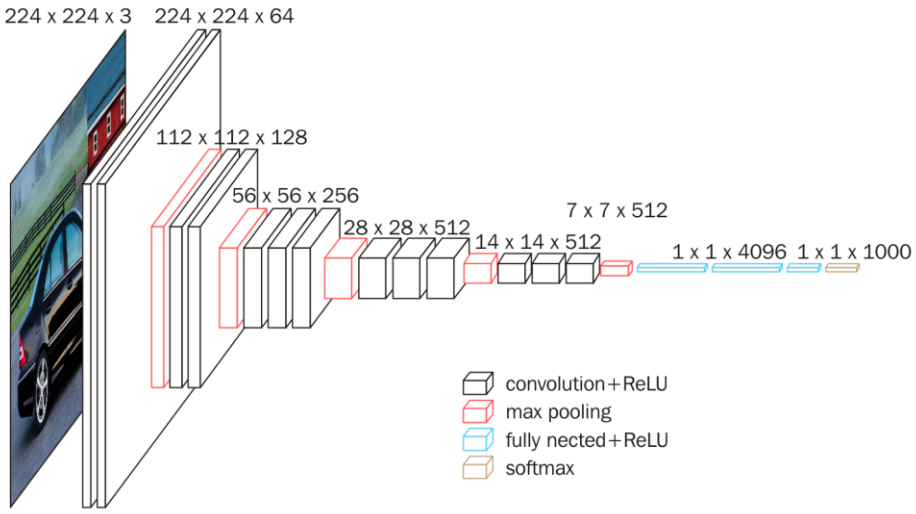
Tabulka 1: Architektura Tiny Yolov3 modelu.

#	Typ vrstvy	Filtr	Velikost	Krok	Vstup	Výstup
1	Konvoluční	16	3x3	1	416x416x3	416x416x16
2	Maxpool		2x2	2	416x416x6	208x208x16
3	Konvoluční	32	3x3	1	208x208x16	208x208x32
4	Maxpool		2x2	2	208x208x32	104x104x32
5	Konvoluční	64	3x3	1	104x104x32	104x104x64
6	Maxpool		2x2	2	104x104x64	52x52x64
7	Konvoluční	128	3x3	1	52x52x64	52x52x128
8	Maxpool		2x2	2	52x52x128	26x26x128
9	Konvoluční	256	3x3	1	26x26x128	26x26x256
10	Maxpool		2x2	2	26x26x256	13x13x256
11	Konvoluční	512	3x3	1	13x13x256	13x13x512
12	Maxpool		2x2	2	13x13x512	13x13x512
13	Konvoluční	1 024	3x3	1	13x13x512	13x13x1024
14	Konvoluční	256	1x1	1	13x13x1024	13x13x256
15	Konvoluční	512	3x3	1	13x13x256	13x13x512
16	Konvoluční	18	1x1	1	13x13x512	13x13x18
17	Yolo					
18	Route					
19	Konvoluční	128	1x1	1	13x13x256	13x13x128
20	Upsample			2x	13x13x128	26x26x128
21	Route					
22	Konvoluční	256	3x3	1	26x26x384	26x26x256
23	Konvoluční	18	1x1	1	26x26x256	26x26x18
24	Yolo					

Dataset byl v tomto případě zvolen z práce [4] určený pro jejich implementaci CNN. Obsahuje 3 000 obrázků v různých scénách. Z těch bylo vybráno 1000 obrázku. Pro využití dat v tomto modelu bylo nutné je speciálně anotovat. Anotace definuje třídu a obdélník ve kterém se objekt nachází. Obdélník je určen poměrem x, y souřadnic, výšky a šířky k hodnotám celého obrázku. Tento druh anotace lépe funguje při použití různých velikostí obrázků.

4.1.3 Model s využitím CNN a sliding window

Na rozdíl od předchozího modelu je cílem statistická přesnost detekce zbraně. Vyzkoušeny byly dvě architektury VGG16 a vlastní navržená architektura. VGG16 (názorně na obrázku 3) již byla předtrénovaná a jen bylo toto trénování doplněno tak, aby vyhovovalo klasifikace zbraní. Vzhledem k tomu, že klasifikátor na bázi CNN potřebuje fixní velikost vstupního obrazu a není známá velikost zbraně v poměru k velikosti obrázku bylo potřebné tento aspekt ošetřit. Podobný přístup jako v modelu na bázi HOG a SVM byl použit (viz kapitola 4.1.1). Na konec modelu byly připojeny dvě plně-propojené vrstvy (první má počet filtrů 1 024 a druhá počet tříd které mají být klasifikovány).

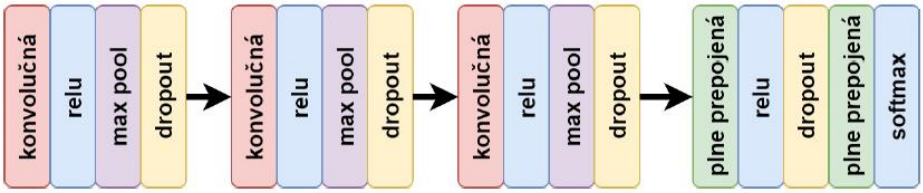


Obrázek 3: Architektura VGG16 [5].

Prvotní návrh architektury je v tabulce 2. Jedná se o tři konvoluční a maxpool vrstvy. Jako aktivační funkce je využito ReLu. Pro zvýšení zobecnění modelu se využívají i dropout vrstvy. Hodnota dropout vrstev byla nastavena na 0,2. Jedna dropout vrstva je vložena i mezi plně-propojené vrstvy (tentokrát s hodnotou 0,5). Kompletní architektura je tak naznačena na obrázku 4.

Tabulka 2: Architektura vlastního návrhu modelu.

#	Typ vrstvy	Filtr	Veli kost	Krok	Vstup	Výstup
1	Konvoluční	32	3x3	1	64x64x3	64x64x32
2	Maxpool		2x2	2	64x64x32	32x32x32
3	Konvoluční	64	3x3	1	32x32x32	32x32x64
4	Maxpool		2x2	2	32x32x64	16x16x64
5	Konvoluční	128	3x3	1	16x16x64	16x16x128
6	Maxpool		2x2	2	16x16x128	8x8x128



Obrázek 4: Architektura vlastního návrhu modelu.

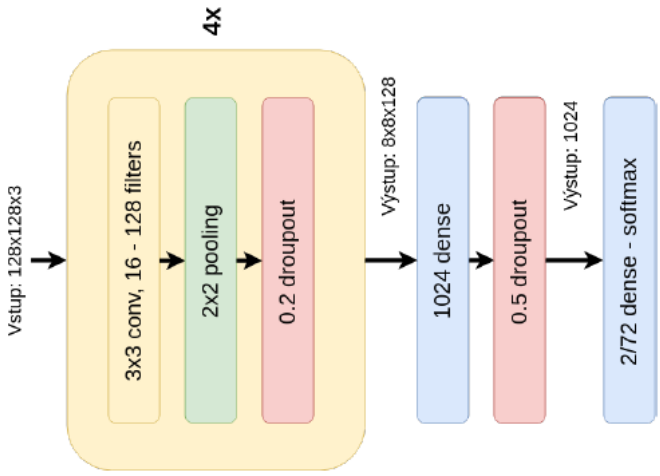
Pro trénování je použitý stejný dataset jako v prvním modelu (HOG a SVM, kapitola 4.1.1). Ten obsahuje 102 tříd 9 857 obrázků a z toho 795 zbraní (krátkých). Velké množství obrázku však obsahuje čistě bílé pozadí. Pro větší robustnost byly přidány i data z [6]. Vhodné data byly manuálně vybrány a přiměřeném rozsahu přidány. Prozkoumán byl i vliv počtu tříd na klasifikaci. Proto byla vyzkoušena varianta se 2 (obsahuje zbraň a neobsahuje zbraň) i původní ze 102 třídami.

4.2 Popis zbraní

Tato část je věnována klasifikaci popisu zbraní. V tomto případě se jedná o délku zbraně a její natočení. Hlavním nástrojem budou konvoluční neuronové sítě. Pro možnost porovnání výsledků budou navrženy dvě sítě. První z nich je inspirovaná architekturou AlexNet (viz kapitola 3.1.2). Model obsahuje 15 vrstev (4 konvoluční, 4 maxpooling, 5 dropout a 2 plně-propojené vrstvy). Velikost vstupních dat do první vrstvy je 128x128x3.

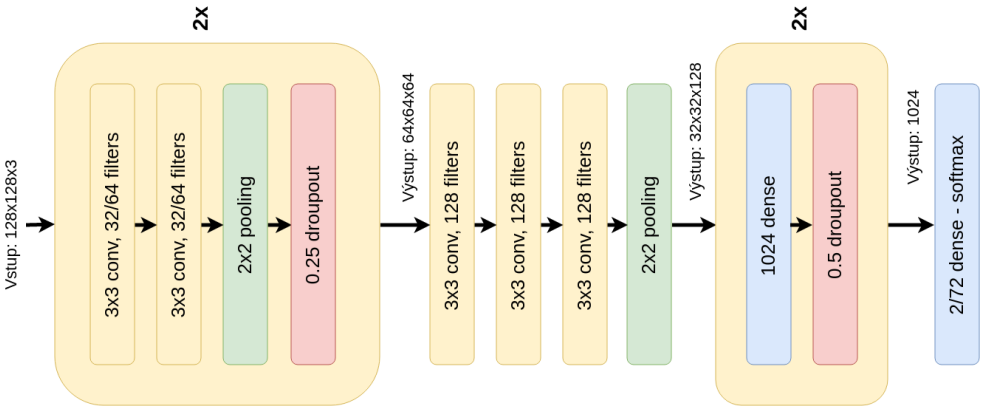
Ve všech konvolučních vrstvách jsou použity filtry velikosti 3x3 s krokem 1 a nulovým zarovnáním. V modelu se s každou konvoluční vrstvou zdvojnásobuje počet filtrů z počátečních 16 až na 128 v poslední vrstvě. Pooling (resp. maxpooling) vrstvy mají velikost filtru 2x2 s posunem 2 v každé ose. Za nimi se pak nachází dropout vrstva s nastavením 0,2 (tzn. 20 % náhodných propojení se ignoruje).

Po 4 blocích konvoluční, pooling, a dropout vrstvy následuje dense vrstva s počtem propojení 1 024 a dropout vrstva s nastavením 0,5. Jako poslední je dense vrstva s 2 nebo 72 propojeními a softmax klasifikátorem. Počet výstupu závisí, jestli se určuje typ nebo náklon zbraně. V celé sítu jsou použité ReLu aktivační funkce. Architektura sítě je přehledně na obrázku 5.



Obrázek 5: AlexNetLike navrhovaná architektura.

Druhý navrhovaný model je inspirovaný architekturou VGG sítí (viz kapitola 3.1.2). Pro zrychlení trénování je síť o dva bloky vrstev menší a konvoluční vrstvy obsahují méně filtrů. Celkově síť obsahuje 2 bloky s 2 konvolučními (počet filtrů 32 a 64), pooling a dropout (0,2) vrstvami. Poté následují 3 konvoluční vrstvy (počet filtrů 128) a pooling vrstva. Jako poslední jsou 2 bloky s dense (počet propojení 2048) a dropout vrstvou (0,5). Poslední výstupní vrstva obsahuje 2 nebo 72 propojení se softmax klasifikátorem. Každá konvoluční vrstva obsahuje filtry o velikost 3x3, krokem 1 a použitím nulového doplňku. Pooling vrstvy jsou typu max, velikost filtru 2x2 a posun 2 po každé ose. V celé síti je použitá ReLu aktivační funkce. Architektura sítě je přehledně na obrázku 6.



Obrázek 6: VGGLike navrhovaná architektura.

4.2.1 Typ (délka) zbraně

Při klasifikaci délky zbraně jsou využity i klasické metody. Pro ně je potřeba předzpracovat vstupní data pomocí konverze do odstínů šedi a HOG. Na tato data bude využitý klasifikátor K-Nearest-Neighbour (KNN, k nejbližších sousedů) a SVM. Ke klasifikaci budou však použity i neuronové sítě z předchozí kapitoly 4.2. Předzpracování dat pro tento přístup bude jen v normalizaci RGB hodnot, nastavení rovnoměrné velikosti stran obrázků a augmentaci dat pro zvětšení počtu vstupních dat.

Dataset pro tuto klasifikaci byl vybrán jako kompilát 3 zdrojů (IMFDB, ImageNet a Google). IMFDB je databáze záběrů z filmů, ve kterých se nacházejí zbraně. Obsahují nejen celé scény, ale i samostatné obrázky zbraní, které se v dané scéně nacházejí. Pro klasifikaci je však potřeba, aby obrázek obsahoval pouze zbraně, a proto bylo potřeba obrázky ručně vyfiltrovat. Následně bylo nutné udělat runě klasifikaci na dlouhé a krátké. ImageNet je databáze obrázků, která obsahuje víc než 14 milionů obrázků ve více než 21 000 kategoriích. Výhodou této databáze je, že už je anotovaná. Pro doplnění a zvětšení počtu obrázků je možno využít i služeb vyhledávání Googlu. Celkový počet využitých obrázku je v tabulce 3.

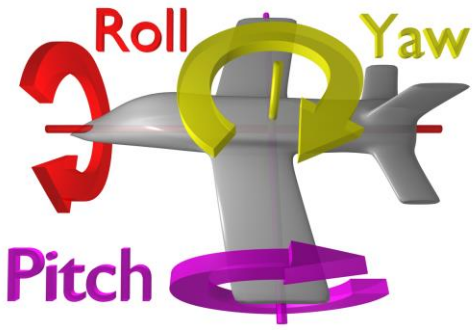
Tabulka 3: Podrobné počty trénovacích dat.

Zdroj	Počet krátkých zbraní	Počet dlouhých zbraní
IMFDB	647	670
ImageNet	730	94
Google	0	86
Dohromady	**1 377**	**850**

Následně probíhá augmentace dat. To je jedna z možností, jak navýšit jejich množství a zlepšit generalizaci. Použité metody jsou: rotace (o úhel 0-180°), překlopení (vertikální nebo horizontální), posun (v ose *x* nebo *y až* o 15 % délky, resp. šířky). Při rotaci obrázků můžou vzniknout místa bez určené barvy v takovém případě je barva těchto „neznámých“ pixelů vyplněna barvou hraničního pixelu.

4.2.2 Orientace

Dalším cílem je určení náklonu zbraně v obraze. Náklon se zjišťuje ve všech třech osách. Pro zjednodušení se využívají pojmy z letectví (roll, yaw a pitch) viz obrázek 7. Klasifikace natočení bude probíhat s využitím CNN (viz kapitola 4.2). Výsledek poslední vrstvy softmax je 72 výstupů, ty určují, o jaký úhel je zbraň natočena. Každá ze 72 tříd zastupuje rozpětí 5 stupňů. Předzpracování je stejné jako při klasifikaci délky. Tzn. normalizace, úprava rozměru vstupu na čtverec a augmentace dat. Pro každý náklon bude natrénovaná samostatná CNN. Přesnost sítě pro určení natočení je průměrný rozdíl mezi skutečnými a klasifikovanými úhly.



Obrázek 7: Pojmenování os pro letectvo [7].

Vhodný dataset pro klasifikaci náklonu je obtížné. Nakonec byla využita databáze Free3D. To je databáze 3D modelů zbraní včetně texturních informací v různých formátech. Tím bylo vyřešeno získání 3D modelů, k tomu, aby z nich vznikli anotované verze různých natočení byl využit SYDAgenerátor. Ten pracuje s 3D modelem a obrázkem pozadí. 3D model umí do pozadí (scény) vložit s využitím jakékoliv natočení ve všech třech osách. Zároveň u toho automatizovaně vytvořit anotaci vytvořené rotace. Využitím 10 pozadí a 5 3D modelů (3 pro dlouhé zbraně a 2 pro krátké) bylo vytvořeno dostatečné množství dat pro otestování přístupu s využitím CNN (viz tabulka 4).

Tabulka 4: Podrobné počty trénovacích dat.

Osy otáčení	Počet obrázků
Pitch	1 480
Roll	4 450
Yaw	4 584

5 Vyhodnocení modelů

V této kapitole jsou vyhodnoceny všechny výše zmíněné modely. Kapitola je rozdělena na detekci a popis zbraní.

5.1 Detekce zbraní

Kapitola obsahuje popis výsledků z modelů pro detekci zbraní – u neuronových sítí také popis trénování dat.

5.1.1 HOG a SVM

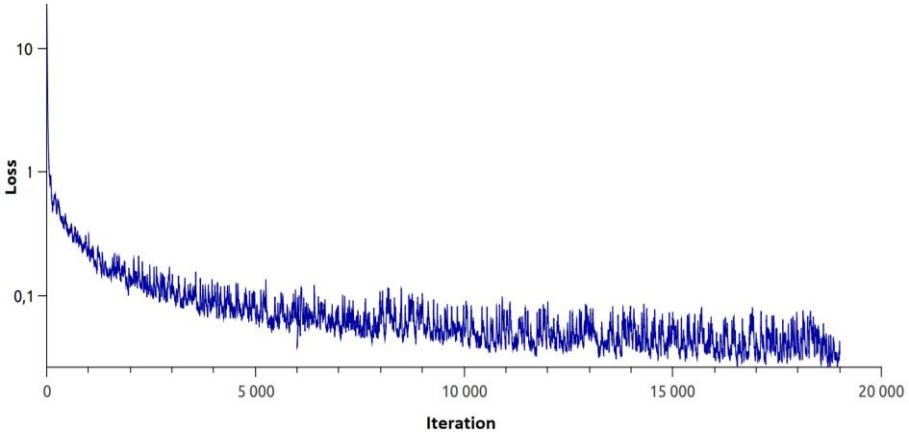
Zdrojové obrázky datasetu použitého při testování zobrazovaly zbraně jako selektivní objekty, tedy s uniformním pozadím a bez přítomnosti čehokoliv jiného v obraze. Pro detekci jsme jako vstupní data použili reálné (filmové) scény, kde se nacházely osoby držící zbraň. Ačkoli model byl schopen ve fázi testování pozitivně

klasifikovat s vysokou mírou pravděpodobnosti, výsledky při detekci nebyly uspokojivé. Vykazoval totiž vysokou míru falešných nálezů, především ve scénách, jako je například obloha. Zbraň byl schopen detekovat zejména podle oblasti spouště, která se však při zbrani, kterou drží ruka nenachází v takovém tvaru, jaký detektor hledal. Toto zjištění považujeme za klíčový důvod, proč tento model nedetekoval zbraň drženou rukou.

Nepodařilo se nám najít žádné nastavení parametrů HOG deskriptoru, které by ve finální fázi detekce reálných obrazů zvoleného datasetu vedlo, nebo se alespoň dostatečně přibližovalo k výsledkům, které bychom mohli považovat za v praxi použitelné. Tento model tak nepovažujeme jako úspěšný ani z hlediska detekčních výsledků, ani z hlediska rychlosti.

5.1.2 Yolov3

Během trénování, jsme ukládali váhy každých 1 000 iterací. Pro každou váhu jsme následně vyhodnotili average precision na testovacích datech. Nejvyšší hodnotou jsme dostali při 19 000 iteracích. Hodnota average precision byla 32,16 %, což je velmi blízko referenční hodnotě, kterou Tiny Yolov3 dosahuje a to 33,1 %. Průběh chybové funkce je na obrázku 8.



Obrázek 8: Průběh chybové funkce při trénovaní Tiny Yolov3.

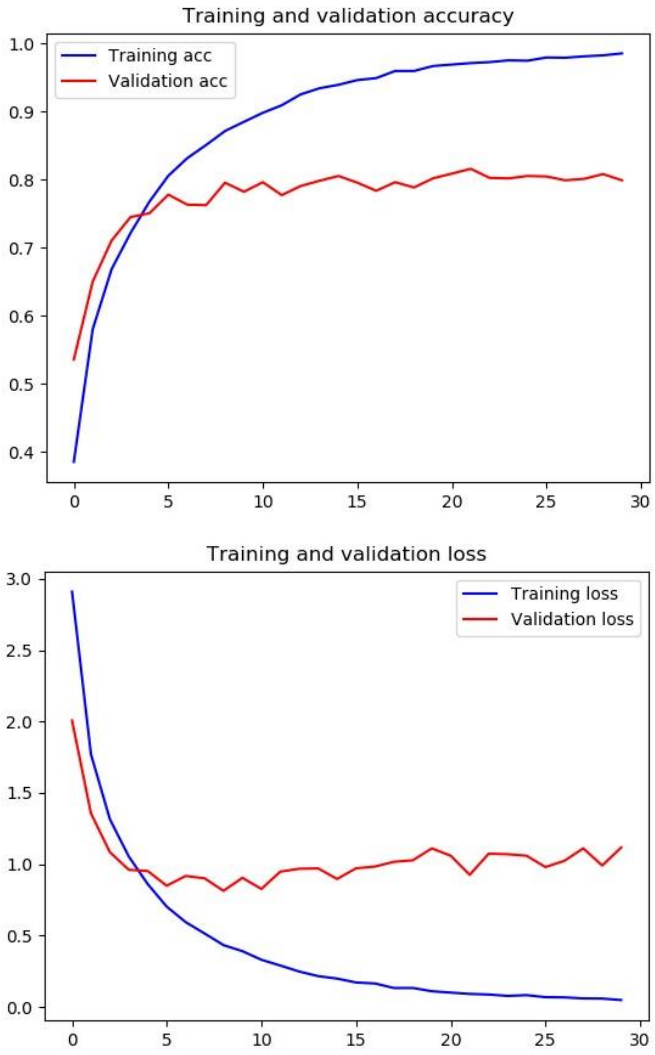
5.1.3 CNN VGG16

Do modelu jsme implementovali klasifikaci do 102 tříd, která výslednou pravděpodobnost váhuje mezi více třídami. Což mělo za následek nižší množství falešně positivních nálezů.

Protože binární klasifikační model na bázi VGG16 dosahoval vyšší průměrné přesnosti, rozhodli jsme se využít právě tento model pro tvorbu klasifikátoru. Průběh trénování můžeme vidět na obrázku 9.

Na testovacích datech dosáhl average precision 97,79 %. Při výsledné detekci tato verze modelu prokazovala mnohem lepší výsledky než předchozí.

Dosažené výsledky a diskuze nad dalším směřování vývoje je taktéž součástí této zprávy.



Obrázek 9: Průběh trénování klasifikace do 102 tříd.

Na vzorku 12 scén, které dohromady obsahovaly 13 zbraní jsme experimentálně určovaly hodnotu prahu. Při hodnotě prahu 0,999 byl počet skutečně positivních nálezů 12, skutečně negativních také 12, a jeden případ falešně negativní detekce.

5.2 Popis zbraní

Přesnost je zde definovaná pomocí tzv. chybové matice. A vychází se vzorce (2), kde TP (true positive), FP (false positive), FN (false negative) a TP (true positive) viz [8]. V tabulce 5 je uvedeno celkové srovnání výsledků pro určení typu zbraně. Z těchto výsledků můžeme vidět, že navrhovaná architektura AlexNetLike dosáhla nejlepší částečné, ale hlavně i celkové úspěšnosti a to až 83.14 %. Barevně jsou

vyznačeny nejlepší (zelené) a nejhorší (červené) výsledky pro jednotlivé přesnosti.

$$Přesnost = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Tabulka 5: Celkové srovnání pro určení typu zbraně.

Metoda	Dlouhá zbraň – přesnost	Krátká zbraň – přesnost	Celková přesnost
SVM	61 %	59 %	**59,65 %**
SVM – linear	57 %	62 %	**59,29 %**
KNN - 1	70 %	69 %	**69,49 %**
KNN - 5	70 %	68 %	**69,22 %**
AlexNetLike	94 %	73 %	**83,14 %**
VGGLike	87 %	48 %	**67,06 %**

Dále tabulka 6 zobrazuje výsledky pro jednotlivé osy a dvě navrhované architektury. Je jasně vidět, že architektura AlexNetLike řádově překonala úspěšností druhé navrhované architektury V hlavičce tabulky je uvedeno p, které označuje hodnotu prahové hodnoty pro určení správné predikce. Barevně jsou znovu vyznačeny nejlepší a nejhorší výsledky.

Tabulka 6: Souhrn srovnání dosažených výsledků pro určení náklonu zbraně.

Osa rotace	Metoda	Přesnost pro p = 5	Přesnost pro p = 10
Pitch	AlexNetLike	**85,14 %**	**92,34 %**
	VGGLike	**4,05 %**	**7,66 %**
Roll	AlexNetLike	**91,02 %**	**95,51 %**
	VGGLike	**3,74 %**	**5,39 %**
Yaw	AlexNetLike	**49,71 %**	**54,65 %**
	VGGLike	**3,78 %**	**5,96 %**

V závěru lze konstatovat, že pro řešené problémy jsou vhodnější menší konvoluční neuronové sítě na rozdíl od těch hlubokých, protože velikost trénovacích dat je velmi malá, řádově jen v pár tisících. Výsledkem tohoto srovnání jsou čtyři nejlepší modely, AlexNetLike pro určení typu zbraně a tři AlexNetLike modely pro určení náklonu zbraně, tyto čtyři modely je možné použít pro celkový popis zbraně v obraze.

6 Závěr

Zpráva popisuje možnosti detekce, klasifikace dle délky a určení natočení zbraní v obrázcích. U detekce zbraní byl vyzkoušen model s HOG a SVM, neuronovou sítí založenou na Yolov3 a sítí založenou na VGG16. Klasické metody (HOG a SVM) propadly, a to jak z pohledu rychlosti, tak z pohledu přesnosti. Model založený na architektuře Tiny Yolov3 měl úspěšnost 32,16 % tj. třetina případů. Smysluplně úspěšný byl až model založený na architektuře VGG, který dosáhl při testovacích datech 97,8 % přesnosti. Při validaci se přesnost pohybovala nad 80 % což je relativně slušný výsledek.

U klasifikace typu (délky) zbraně byl znovu vyzkoušen přístup klasických metod (HOG a SVM) a (HOG a KNN), i přístup pomocí neuronových sítí. První založený na VGG a druhý založený na AlexNet. I u tohoto problému měly klasické metody špatné výsledky. Konkrétně SVM kolem 59 % tedy o něco lepší než náhodný tip výsledku. Metoda založena na KNN pak dosáhla 69 % úspěšnosti. Síť založena na architektuře VGG (která se osvědčila při detekci) dosáhla pouhých 67 %. Pořádný výsledek nakonec měla jen sít založena na AlexNet architektuře s 84,14 % přesnosti.

Posledním částí je určení orientace zbraně. U toho přístupu nebyly vyzkoušeny klasické metody, ale pouze sítě založené na VGG a AlexNet architektuře. Za zmínku také stojí automatizované vygenerování dat za pomocí SYDAgenerátoru. VGGLike síť dosáhla násobně horších výsledku než AlexNet (přesnost se pohybovala kolem 3,8 %). AlexNetLike síť měla velmi dobré výsledky u pitch (85,14 %) a roll (91,02 %). Nicméně u yaw byl výsledek pouze (49,71 %) což ukazuje větší náročnost detekce tohoto druhu natočení.

Závěrem lze tedy říct, že se osvědčili pouze přístupy využívající NN. Z pohledu konkrétních sítí má smysl dále zkoumat hlavně síť s architekturou AlexNet. Zároveň se osvědčilo získávání dat za pomocí SYDAgenerátoru, který s relativně malým počtem počátečních obrázků (resp. 3D modelů) vygeneroval slušný dataset.

Literatura

[1] Redmon, J.; Divvala, S. K.; Girshick, R. B.; aj.: You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: s. 779–788.

[2] Kevin, M.: Simple guide to confusion matrix terminology. online, [Online; navštívené 07.05.2018]. URL http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

[3] B, N.: Image Data Pre-Processing for Neural Networks. online, [Online; navštívené 10.01.2018]. URL https://becominghuman.ai/image-data-pre-processing-for-neuralnetworks-498289068258

[4] Hosang, J.; Benenson, R.; Dollár, P.; aj.: What makes for effective detection proposals? IEEE transactions on pattern analysis and machine intelligence, 2016: s. 814–830.

[5] Girshick, R.; Donahue, J.; Darrell, T.; aj.: Rich feature hierarchies for accurate object detection and semantic segmentation. UC Berkeley, 2014.

[6] Aircraft principal axes. online, [Online; navštívené 09.05.2018]. URL https://en.wikipedia.org/wiki/Aircraft_principal_axe

[7] Internet Movie Firearms Database. [Online; navštívené 11.05.2018]. URL http://www.imfdb.org/wiki/Main_Page

[8] Rohit Kumar Tiwari, G. K.: A Computer Vision based Framework for Visual Gun Detection Using Harris Interest Point Detector. [Online; navštívené 11.05.2018]. URL https://www.sciencedirect.com/science/article/pii/S1877050915014076