



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

PASIVNÍ OPTICKÁ DETEKCE A KLASIFIKACE LETÍCÍCH OBJEKTŮ

PASSIVE OPTICAL DETECTION AND CLASSIFICATION OF FLYING OBJECTS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Andrea Mošková

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Eliška Vlachová Hutová, Ph.D.

BRNO 2022

Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Studentka: Andrea Mošková

ID: 221005

Ročník: 3

Akademický rok: 2021/22

NÁZEV TÉMATU:

Pasivní optická detekce a klasifikace letících objektů

POKyny PRO VYPRACOVÁNÍ:

1. Zpracujte videonahrávky z bezpečnostních kamer v blízkosti letiště za účelem následného zpracování obrazů.
2. Využijte nejvhodnější metodu, která byla teoreticky popsána v semestrálním projektu. Pomocí této metody vytvořte detekční algoritmus, který rozpozná letící objekt a dokáže ho klasifikovat.
3. Výstupem práce bude informace o detekovaném objektu: typ daného objektu a jeho souřadnice v prostoru.

DOPORUČENÁ LITERATURA:

[1] YOSHIHASHI, R., R. KAWAKAMI, M. IIDA a T. NAEMURA. Bird detection and species classification with time-lapse images around a wind farm: Dataset construction and evaluation. Wind Energy [online]. 2017, 20(12), 1983-1995 [cit. 2021-3-8]. ISSN 10954244. Dostupné z: doi:10.1002/we.2135

[2] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. Image processing, analysis, and machine vision. 4th edition. United States of America: Cengage Learning, 2015. International Edition. ISBN 978-1-133-59369-0.

Termín zadání: 7.2.2022

Termín odevzdání: 23.5.2022

Vedoucí práce: Ing. Eliška Vlachová Hutová, Ph.D.

doc. Ing. Václav Jirsík, CSc.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Na základe úvodného rozboru metód využívaných pre optickú detekciu a rozpoznávanie pohybujúcich sa elementov v dynamickom obraze snímanom statickou kamerou, predstavuje bakalárska práca návrh komplexného detekčného algoritmu, schopného vo videonahrávkach zaznamenať a klasifikovať letiace objekty na oblohe. Metóda využíva diferenčný algoritmus pre detekciu pohybových príznakov, vďaka ktorým sú potom v snímkach videosekvencie lokalizované letiace objekty. SVM klasifikátor následne rozhodne o zaradení každého objektu do jednej z troch uvažovaných tried, a to na základe jeho popisu získanom extrakciou SIFT deskriptorov a prípadne doplnenom aj o informácie z RGB histogramov. Algoritmus bol implementovaný primárne v programe Matlab.

Kľúčové slová

Detekcia pohybu, rozpoznávanie objektov, strojové učenie, SIFT, SVM, Matlab.

Abstract

Based on an initial analysis of the methods used for optical detection and recognition of moving elements in a dynamic image captured by a static camera, the bachelor's thesis presents a proposal for a complex detection algorithm capable of recording and classifying flying objects in the sky. The method uses a differential detection algorithm to detect signs of motion, thanks to which flying objects are located in the video frames. The SVM classifier then decides on the classification of the object into one of the three considered classes, based on its description obtained by extraction of SIFT descriptors and possibly supplemented with information from RGB histograms. The algorithm was implemented primarily in Matlab.

Keywords

Motion detection, object recognition, machine learning, SIFT, SVM, Matlab.

MOŠKOVÁ, Andrea. *Pasivní optická detekce a klasifikace letících objektů*. Brno, 2022. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/142710>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Eliška Vlachová Hutová.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta: *Andrea Mošková*

VUT ID studenta: *221005*

Typ práce: *Bakalářská práce*

Akademický rok: *2021/22*

Téma závěrečné práce: *Pasivní optická detekce a klasifikace letících objektů*

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 20. května 2022

podpis autora

Pod'akovanie

Rada by som pod'akovala vedúcej bakalárskej práce pani Ing. Eliške Vlachovej Hutovej, Ph.D. za užitočnú odbornú i metodickú pomoc, ochotný a trezlivý prístup aj prínosné konzultácie.

V Brne dňa: 20. mája 2022

podpis autora

Obsah

ZOZNAM OBRÁZKOV	9
ZOZNAM TABULIEK	11
ÚVOD	12
1. PASÍVNA OPTICKÁ DETEKCIA	13
1.1 POŽIADAVKA INVARIANTNOSTI.....	13
2. SPRACOVANIE OBRAZU.....	15
2.1 PROBLEMATIKA POČÍTAČOVÉHO VIDENIA	15
2.2 POSTUP SPRACOVANIA OBRAZU	16
2.3 PREDSPRACOVANIE	17
<i>Bodové jasové transformácie</i>	<i>17</i>
<i>Geometrické transformácie</i>	<i>17</i>
<i>Lokálne predspracovanie</i>	<i>18</i>
<i>Obnovenie obrazu</i>	<i>18</i>
2.4 SEGMENTÁCIA.....	18
3. METÓDY DETEKČIE A KLASIFIKÁCIE POHYBUJÚCICH SA OBJEKTOV	20
3.1 DETEKČIA POHYBU V SEKVENCII OBRAZOV.....	20
<i>Optický tok</i>	<i>20</i>
<i>Diferenčná metóda</i>	<i>21</i>
<i>Metóda odčítania pozadia (Background Subtraction)</i>	<i>21</i>
3.2 PREHLAD MOŽNOSTÍ ROZPOZNÁVANIA POHYBUJÚCICH SA OBJEKTOV.....	22
3.3 IDENTIFIKÁCIA OBJEKTOV NA ZÁKLADE VÝZNAMNÝCH BODOV – SIFT	23
4. DÁTA VYUŽÍVANÉ PRI VÝVOJI METÓDY	28
5. URČOVANIE POLOHY OBJEKTU V OBRAZE	29
5.1 ZISŤOVANIE OBRAZOVÝCH SÚRADNÍC OBJEKTU	29
5.2 ZHODNOTENIE POSTUPU	32
6. NÁVRH KLASIFIKAČNÉHO ALGORITMU	34
6.1 DATASET.....	34
6.2 PARAMETRE PRE EVALUÁCIU MODELOV.....	35
6.3 PRIEBEH UČENIA KLASIFIKÁTORA	37
6.4 POSILNENIE PRÍZNAKOVÝCH VEKTOROV – INFORMÁCIE O FARBE	39
6.5 VZÁJOMNÉ POROVNANIE VÝSLEDNÝCH MODELOV SVM	40
7. TESTOVANIE NAVRHNUTÝCH METÓD DETEKČIE A KLASIFIKÁCIE.....	46
7.1 VÝSTUPY ANALÝZY VIDEOZÁZNAMOV.....	46
7.2 ČASOVÁ ANALÝZA	50
7.3 ZHODNOTENIE VÝSLEDKOV.....	51
ZÁVER	53
LITERATÚRA.....	55

ZOZNAM SKRATIEK	59
ZOZNAM PRÍLOH.....	60

ZOZNAM OBRÁZKOV

Obr. 2.1 „Pinhole“ model geometrie [7]. A) Nasnímaný obraz, B) Malý otvor, C) Virtuálny obraz, D) Skutočný snímaný objekt	15
Obr. 3.1: Mierkovo nezávislá reprezentácia obrazu scale-space s konštrukciou rozdielových snímok $D(x,y,\sigma)$ [21]. A) 2.oktáva scale-space, B) 1.oktáva scale-space. Šípka značí zmenu mierky σ	25
Obr. 3.2: Vyhľadávanie lokálnych extrémov na základe porovnania bodu X s označenými bodmi v jeho okolí [21]. Šípka značí zmenu mierky σ	25
Obr. 3.3: Vypočítané gradienty v okolí významného bodu (vľavo) a histogram (vpravo) s dominantnými orientáciami (červená a zelená šípka) [21].	26
Obr. 3.4: Výsledný deskriptor významného bodu (vpravo) zostavený z gradientov v okolí (vľavo) [21].	27
Obr. 5.1: Postup lokalizácie letiaceho objektu v obraze.	29
Obr. 5.2: Príklad binárnej masky použitej pre elimináciu pohybových príznakov z okolia.	30
Obr. 5.3: Invertovaná binárna rozdielová snímka pred (vľavo hore) a po úpravách (vpravo hore), finálna lokalizácia letiaceho objektu v snímke z videosekvencie (dolu).	31
Obr. 5.4: Invertované zobrazenie hrán pred (vľavo hore) a po úpravách (vpravo hore), finálna detekcia letiacich objektov v samostatnej snímke (dolu). Pre prehľadnosť nie sú zobrazené súradnice.	32
Obr. 5.5: Invertovaná binárna rozdielová snímka pred (vľavo hore) a po úpravách (vpravo hore), finálna lokalizácia letiaceho objektu na oblohe aj pohybujúcich sa objektov v okolí, ktoré sa nepodarilo odfiltrovať maskou (dolu). Pre prehľadnosť nie sú zobrazené súradnice.	33
Obr. 6.1: Ukážky obrázkov z množiny snímok pre učenie klasifikátora. V prvom riadku sú zobrazené reprezentácie lietadiel, v druhom vtákov.....	35
Obr. 6.2: Postup navrhutej metódy klasifikácie na základe SIFT deskriptorov	39
Obr. 6.3: Priebeh klasifikácie realizovanej na základe SIFT deskriptorov aj RGB histogramov	40
Obr. 6.4: ROC pre triedu lietadiel (<i>Plane</i>).....	41
Obr. 6.5: ROC pre triedu vtákov (<i>Bird</i>).....	41
Obr. 6.6: ROC pre triedu <i>Negative</i>	42
Obr. 6.7: ROC pre triedu lietadiel (<i>Plane</i>) - testovacia množina	43
Obr. 6.8: ROC pre triedu <i>Negative</i> - testovacia množina.....	44
Obr. 6.9: ROC pre triedu vtákov (<i>Bird</i>) - testovacia množina	44
Obr. 7.1: Postup analýzy videozáznamov	46
Obr. 7.2: Správna detekcia lietadla (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).	47
Obr. 7.3: Správna detekcia vtáka (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).	47
Obr. 7.4: Príklad spracovania samostatnej fotografie mimo videosekvencie - korektná detekcia vtákov (vpravo hore), originálna snímka (vľavo hore), pôvodná binárna hranová reprezentácia (vľavo dole), maskou filtrovaná hranová snímka (vpravo dole).	48
Obr. 7.5: Nesprávne rozpoznanie vtáka ako lietadlo (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).....	49
Obr. 7.6: Chybné rozpoznanie lietadla ako vtáka (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).	49

Obr. 7.7: Pohybujúce sa objekty náležiacie triede <i>Negative</i> (vľavo listy, vpravo oblačnosť) nesprávne označené ako lietadlá a vtáky.....	50
Obr. 7.8: Porovnanie modelu klasifikátora využívajúceho iba SIFT deskriptory (vľavo) s modelom klasifikujúcim na základe SIFT aj RGB príznakov (vpravo).	52

ZOZNAM TABULIEK

Tabuľka 6.1: Početnosť reprezentácií jednotlivých tried v rámci tréningovej množiny	35
Tabuľka 6.2: Matica zámen	35
Tabuľka 6.3: Úspešnosť klasifikátorov spracúvajúcich SIFT deskriptory.....	38
Tabuľka 6.4: Úspešnosť klasifikátorov spracúvajúcich popisy snímok tvorené SIFT deskriptormi aj RGB histogramami.....	40
Tabuľka 6.5: Vypočítané hodnoty AUC pre vybrané naučené modely SVM.....	42
Tabuľka 6.6: Overenie výsledkov vybraných SVM modelov na testovacej množine	43
Tabuľka 6.7: Vypočítané hodnoty AUC – overovanie úspešnosti vybraných SVM modelov na testovacej množine.....	45
Tabuľka 7.1: Potrebná doba spracovania jednej snímky z videosekvencie, na ktorej bol zaznamenaný pohyb.....	51

ÚVOD

Bakalárska práca sa zaoberá možnosťami detekcie letiacich objektov na oblohe v dynamickom obraze získanom staticky upevnenou kamerou, určovaním ich súradníc v uvažovanej snímke a rozpoznávaním daného objektu. Z hľadiska zvýšenia informačnej hodnoty výstupov celého postupu by bolo vhodné zamerať sa aj na určovanie GPS polohy spozorovaného letiaceho prvku, avšak snímacie zariadenia uvažované v rámci práce nie sú dostatočne technicky pripravené na takúto úlohu. Práca sa preto obmedzuje na skúmanie pozície objektu z hľadiska umiestnenia zodpovedajúcich pixelov v obraze.

Cieľom bakalárskej práce je návrh a testovanie komplexného detekčného algoritmu, ktorý by bolo možné podľa potreby vhodne adaptovať a implementovať v rámci rôznych aplikácií. Uplatnenie by sa tak dalo nájsť napríklad pri sprostredkovaní bezpečnostnej analýzy letiacich objektov snímaných v okolí letísk či v identifikácii objektov v okolí vinohradov, ovocných sádov alebo súkromných záhrad s prípadným využitím v systémoch určených na plašenie škodcov na úrode.

V prvej časti bakalárskej práce je stručne vysvetlená problematika pasívnej optickej detekcie a požiadavky kladené na algoritmy, ktoré ju realizujú. Ďalej sú v druhej kapitole rozoberané základné možnosti úpravy obrazov, ktoré umožňujú následnú analýzu, pričom sú detailnejšie predstavené najmä postupy predspracovania a segmentácie obrazu. Tretia časť práce je venovaná teoretickému rozboru možností zaznamenania pohybujúcich sa elementov v sekvencii po sebe nasledujúcich snímok a metód ich klasifikácie. Pozornosť je kladená primárne na postupy využiteľné v rámci uvažovanej témy snímania oblohy a letiacich objektov. Detailne je popísaná metóda zvolená pre praktickú implementáciu v rámci tejto práce. Samostatná kapitola sa venuje predstaveniu obrazových dát využívaných pri vývoji detekčného nástroja a zároveň vysvetľuje postup extrakcie pohybových informácií z týchto obrazov. Piata časť práce zoznamuje s navrhnutým postupom lokalizácie letiacich objektov v jednotlivých snímkach videosekvencie na základe pohybových príznakov. V šiestej kapitole je predstavený vývoj klasifikátora založeného na strojovom učení, schopného zaradiť zaznamenané objekty v obraze do zvolených tried. Navrhnutých, testovaných a vzájomne porovnávaných bolo viacero modelov. Nakoniec je fungovanie celého algoritmu posudzované prostredníctvom analýzy dostupných videozáznamov a bilancie dosiahnutých výsledkov.

1. PASÍVNA OPTICKÁ DETEKCIA

Optická detekcia pohybujúcich sa objektov predstavuje dôležitú úlohu z oblasti počítačového videnia a spracovania obrazu, ktorá nachádza široké možnosti aplikácií. Uplatní sa napríklad pri monitorovaní dopravnej situácie či bezpečnosti na letiskách, pri strážení objektov a území v rámci bezpečnostných systémov alebo môže predstavovať podporu environmentálnych projektov a výskumov.

Príklady využitia optickej detekcie zameranej na letiace objekty, ktorej sa táto práca venuje primárne, možno nájsť pri sledovaní vzdušného priestoru v okolí letísk, v rámci vojenskej a obrannej techniky, monitorovaní vtáctva popísanom v článku [1] alebo ochrane viníc pred škodcami, ktorá bola cieľom práce [2].

Pri riešení úlohy detekcie a klasifikácie objektu v pohybe ide najmä o spracovanie získaného digitalizovaného obrazu z kamery, ktoré bude v kontexte tejto práce pozostávať z nasledujúcich krokov:

- 1) detekcia pohybu v obraze,
- 2) vymedzenie oblasti výskytu objektu, ku ktorému sa zaznamenaný pohyb vzťahuje a jeho vyseparovanie,
- 3) klasifikácia objektu,
- 4) signalizácia rozpoznania objektu.

Pre zabezpečenie možnosti širšieho využitia výsledkov detekcie je vhodné, aby použitá metóda poskytovala údaj o polohe zaznamenaného objektu v priestore alebo v obraze, na čo sa zameria aj táto práca.

1.1 Požiadavka invariantnosti

Náročnosť úlohy optickej detekcie objektov výrazne zvyšujú nepriaznivé odchýlky podmienok, v ktorých je realizovaná, vznikajúcich vplyvom prostredia, postupom získavania obrazu či charakterom samotných zaznamenaných objektov. Efektívny algoritmus by mal byť v závislosti na aplikácii schopný čeliť nasledujúcim javom [3], [4]:

- a) Vplyv osvetlenia: vzhľad objektu závisí od intenzity, uhlu aj farby osvetlenia. Mal by však byť rozoznatelný bez ohľadu na zmeny v osvetlení.
- b) Mierka: prezentácia rovnakého objektu zaznamenaného v rôznej vzdialenosti od zariadenia zachytávajúceho obraz je odlišná. Algoritmy by mali rozdiely v mierke kompenzovať.
- c) Rotácia: objekt môže byť kamerou zaznamenaný v rotácii, čo by však nemalo ovplyvniť jeho detekciu a správnu klasifikáciu.

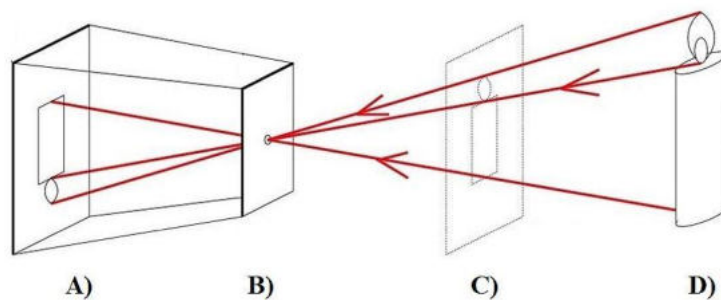
- d) Variácie pozadia: zachytený obraz nezobrazuje iba samotný objekt, ale nesie aj informácie týkajúce sa pozadia, ktoré má často rozmanitý a premenlivý charakter. Rozpoznávanie by však vplyvom zmien pozadia nemalo byť narušené. Napríklad v prípade detekcie letiacich objektov na oblohe by metóda nemala byť citlivá na premenlivú oblačnosť, či výskyt tieňov.
- e) Čiastočný výskyt: Algoritmus by sa mal vysporiadať aj s prípadom, že sa objekt na snímke nevyskytuje celý alebo je čiastočne prekrytý inými predmetmi.
- f) Zmena uhlu pohľadu: vzhľadom k tomu, že pri získavaní obrazu dochádza k transformácii trojrozmerného obrazu na dvojrozmerný, závisí výsledné zobrazenie na relatívnej pozícii kamery voči pozorovanému objektu. Hovoríme teda o uhle pohľadu, ktorý je však premenlivý a vopred neznámy. Algoritmus by mal zaistiť aspoň čiastočnú invariantnosť voči určitému rozsahu zmien uhlu pohľadu.

2. SPRACOVANIE OBRAZU

Úloha optickej detekcie pohybujúcich sa objektov je postavená na spracovaní obrazu, súvisiacim s dynamicky sa rozvíjajúcim oborom počítačového videnia. Táto rozsiahla disciplína sa pokúša o napodobovanie procesu ľudského videnia. Zaznamenávanie obrazu biologickým senzorm nahrádza elektronickými prostriedkami a následne sa počítačovým spracovaním s využitím umelej inteligencie snaží dospieť k čo najkvalitnejšej analýze, porozumeniu a interpretácii obrazu, ktoré sú v prípade ľudského vnímania zabezpečené činnosťou mozgu a ovplyvnené inteligenciou či skúsenosťami [5], [6].

2.1 Problematika počítačového videnia

Počítačové videnie sa vo svojej komplexnosti a náročnosti stretáva s viacerými problémami a obmedzeniami. Patrí k nim napríklad strata informácií pri transformácii z 3D na 2D zobrazenie daná tzv. „pinhole“ modelom geometrie (Obr. 2.1) používanej pri zachytávaní obrazu, ktorá neberie do úvahy veľkosť a vzdialenosť snímaného objektu.



Obr. 2.1 „Pinhole“ model geometrie [7]. A) Nasnímaný obraz, B) Malý otvor, C) Virtuálny obraz, D) Skutočný snímaný objekt

Značnú výzvu pre počítačové videnie predstavuje aj jeho významná úloha interpretácie obrazu, ktorá je v porovnaní s ľudskou schopnosťou porozumieť obrazu, realizovanou nevedomky na základe získaných znalostí a skúseností, kvôli rozmanitosti okolia veľmi zložitá a limitovaná.

Náročnosť procesu spracovania a interpretácie obrazu zvyšuje aj vplyv šumu, prítomný v každom meraní v rámci reálneho sveta, ktorý je nutné vhodnými metódami korigovať, ale aj závislosť nameraného jasú skúmanej snímky od mnohých faktorov, akými sú osvetlenie, pozícia pozorovateľa, tvar či odrazivosť povrchu.

V rámci počítačového videnia zohráva dôležitú úlohu aj problematika veľkého množstva dát, najmä v prípade spracovávania videonahrávok. Vníname nutnosť zaoberať sa otázkou kompromisu medzi požiadavkami na kvalitu vstupných dát a dôrazom kladeným na efektivitu a rýchlosť aplikácií, z ktorých mnohé vyžadujú prácu v reálnom čase.

Ďalším problematickým aspektom procesu porozumenia obrazu je lokálny prístup k analýze. Spracovávaný je v danom momente iba určitý výsek dát, čo však výrazne komplikuje pochopenie globálneho kontextu [5].

2.2 Postup spracovania obrazu

Proces porozumenia obrazu počítačom si vyžaduje identifikáciu vzťahov medzi vstupným obrazom a uvažovaným modelom reálneho sveta. V konečnom dôsledku sa vstupná informácia redukuje na informáciu relevantnú pre konkrétny kontext aplikácie. Postup spracovania je rozdelený do viacerých krokov a úrovní reprezentácie obrazu, medzi ktorými sú úzke väzby a často ťažko definovateľné hranice, pretože sa navzájom ovplyvňujú. Celý proces možno klasifikovať do dvoch hlavných celkov: nižšia úroveň (*low-level image processing*) a vyššia úroveň spracovania obrazu (*high-level image processing*) [5].

Úrovne spracovania obrazu [8]:

Nižšia úroveň:

- 1) Snímanie, digitalizácia a uloženie obrazu
- 2) Predspracovanie
- 3) Segmentácia

Vyššia úroveň:

- 4) Popis
- 5) Klasifikácia, porozumenie obsahu

Nižšia úroveň spracovania zahŕňa získanie obrazu a jeho digitalizáciu, potrebnú pre ďalšie kroky, ktorá zabezpečí prevedenie vstupného analógového signálu do diskrétného tvaru. Spracovávané obrazové dáta sú vďaka nej reprezentované formou matíc, ktorých prvky – pixely predstavujú jas alebo v prípade farebného snímania niekoľko spektrálnych zložiek (červená, zelená, modrá) na príslušnej pozícii v obraze. Aplikované metódy nepracujú so znalosťou obsahu obrazu, ale zodpovedajú za vhodné úpravy, ktoré napomôžu porozumeniu obsahu vo vyššej úrovni.

Vyššia úroveň spracovania má za cieľ porozumieť obsahu obrazu, pričom sa široko uplatnia metódy umelej inteligencie. Snaží sa o imitáciu ľudského rozpoznávania a schopnosti rozhodovania sa na základe informácie získanej z obrazu [5].

2.3 Predspracovanie

Predspracovanie obrazu zahŕňa operácie na najnižšej úrovni abstrakcie – vstupmi aj výstupmi sú hodnoty intenzity obrazu. Nedokáže zabezpečiť zvýšenie informačného obsahu snímky, naopak, väčšinou ho redukuje. Cieľom predspracovania je úprava obrazových dát, ktorá by mala viesť k potlačeniu nežiaducich skreslení a deformácií vzniknutých pri snímaní či digitalizácii, ale aj zvýraznenie dôležitých črt v obraze, čo uľahčí následné použitie metód spracovania a analýzy [5].

Metódy predspracovania je možné zatriediť do štyroch kategórií, podľa veľkosti okolia pixelu využitého na určenie výstupnej úrovne intenzity daného pixelu [5]:

- a) Bodové jasové transformácie
- b) Geometrické transformácie
- c) Lokálne predspracovanie
- d) Obnovenie obrazu

Bodové jasové transformácie

Metódy modifikujú jasovú hodnotu pixelu a delia sa do dvoch skupín podľa závislosti od jeho polohy v obraze.

Jasové korekcie závislé od pozície pixelu sa snažia o potlačenie systematických degradácií daných nerovnomerným osvetlením objektu pri snímaní, samotným zachytávaním obrazu a digitalizáciou. Využívajú znalosť koeficientu odchýlok od ideálnej prevodovej charakteristiky každého pixelu.

Metódy nezávisiace od pozície pixelu v obraze priradia na základe transformačnej funkcie T každému pixelu novú jasovú hodnotu z intervalu $[f_0 \ f_k]$ vychádzajúc z pôvodnej hodnoty z rozmedzia $[g_0 \ g_k]$:

$$f = T(g) \quad (2.1)$$

Typickým príkladom transformačnej funkcie je prahovanie, ktoré na základe zvolenej hodnoty prahu vedie k vzniku binárneho čiernobieleho obrazu [5].

Geometrické transformácie

Geometrické transformácie eliminujú geometrické skreslenia vyskytujúce sa pri zachytávaní obrazu. Pozostávajú z dvoch základných krokov [5]:

- mapovanie pozície pixelu zo súradníc v pôvodnom obraze na novú pozíciu vo výstupnom obraze. Dôležitými transformáciami sú rotácia, zmena mierky (zmenšenie, zväčšenie) alebo skosenie.

- úprava jasovej hodnoty, ktorej výpočet je zvyčajne daný interpoláciou jasových hodnôt pixelov v blízkom okolí.

Lokálne predspracovanie

Na určenie novej jasovej hodnoty pixelu využívajú operácie lokálneho predspracovania jeho blízke okolie a postupujú na základe žiadaného cieľa, ktorý determinuje ich rozdelenie na vyhladzovacie operácie filtrujúce šum a na metódy zameriavajúce sa na detekciu hrán.

Pri odstraňovaní šumu v obraze sa uplatní konvolúcia, teda výpočet výsledného jasu v určitom bode na základe lineárnej kombinácie jasu v okolí tohto bodu vo vstupnom obraze s váhovými koeficientami danými konvolučnou maskou.

Najjednoduchším prípadom je priemerovanie, ktoré určí výslednú jasovú úroveň pixelu ako priemer hodnôt v okolí. Ďalšou možnosťou je využitie konvolučnej masky, ktorej prvky sú dané Gaussovou funkciou. Efektívne sa uplatní pri potlačení Gaussovského šumu. Spomínané metódy však spôsobujú rozmazávanie hrán v obraze.

Vyhladenie využitím filtra s mediánom priradí bodu v obraze jasovú hodnotu zodpovedajúcu mediánu z jeho okolia a pri tom zabezpečí aj zachovanie ostrých hrán.

Hrany v obraze sú charakterizované náhlymi výraznými zmenami jasovej funkcie, ktoré môžu byť detegované využitím gradientných operátorov. Pri popise hrany je braná do úvahy jej veľkosť a smer. Väčšina gradientných operátorov môže byť vyjadrená pomocou konvolučnej masky [5].

Obnovenie obrazu

Obnovením obrazu sa rozumie snaha o potlačenie porúch v obraze, využitím znalostí o ich pôvode. Degradácia obrazu môže byť spôsobená napríklad chybnými optickými šošovkami, relatívnym pohybom medzi objektom a kamerou, nesprávnym zaostrením alebo atmosférickými turbulenciami.

Cieľom obnovenia je rekonštrukcia originálneho obrazu na základe degradovanej verzie. Príkladom metódy obnovenia obrazu je inverzné filtrovanie, založené na predpoklade, že degradácia bola spôsobená lineárnou funkciou [5].

2.4 Segmentácia

V procese analýzy a porozumenia obrazu zohráva podstatnú úlohu segmentácia, ktorej cieľom je rozdelenie obrazu na časti úzko prepojené s reálnymi predmetmi a oblasťami. Rozlišujeme kompletnú segmentáciu, ktorej výstupom je rozčlenenie na oblasti zodpovedajúce objektom vo vstupnom obraze a čiastočnú segmentáciu rozdeľujúcu obraz na viacero oblastí, ktoré sú charakterizované ako jednotné na základe zvolených vlastností ako jas, farba, odrazivosť či textúra.

Významný prínos segmentácie tkvie v redukcii spracovávaného objemu dát. Avšak detailná a správna segmentácia si obvykle vyžaduje aj spoluprácu s metódami z vyšších

úrovni spracovania obrazu, ktoré využívajú aj špecifické znalosti týkajúce sa obsahovej informácie obrazu. Nejednoznačnosť obrazových dát v kombinácii s často sa vyskytujúcim informačným šumom je jednou z podstatných komplikácií procesu segmentácie.

Na základe dominantných rysov, na ktoré sa segmentačné metódy upriamujú je možné ich rozdeliť do troch kategórií:

- a) Segmentácia prahovaním
- b) Segmentácia na základe určovania hraníc oblastí
- c) Segmentácia narastaním oblastí.

Prahovanie predstavuje najjednoduchšiu segmentačnú metódu, ktorá je časovo aj výpočtovo nenáročná. Vychádza z predpokladu, že mnohé objekty v obraze sa vyznačujú konštantnou odrazivosťou či mierou absorpcie svetla povrchom, ktorými sa odlišujú od pozadia. Vďaka tomu dokáže vhodne zvolená jasová hodnota - prah, zabezpečiť separáciu objektov od pozadia. Metóda môže byť aplikovaná globálne na celý obraz alebo formou lokálneho prahovania na jeho vybrané časti.

Metódy určujúce hranice oblastí sú založené na výstupoch hranových operátorov, zodpovedných za zvýrazňovanie hrán na úrovni predspracovania obrazu. Detegované hrany sú v rámci segmentácie spájané do reťazcov, ktoré popisujú hranice oblastí v obraze. Jednoduchým prístupom eliminujúcim nevýrazné hrany je prahovanie obrazu spracovaného hranovými operátormi. Ďalšou možnosťou je metóda relaxácie hrán, ktorá sa snaží o vytváranie kontinuálnych hraníc uvažujúc vlastnosti hrán v kontexte zvoleného okolia. Hranice sú konštruované na základe prítomnosti a veľkosti hrán v okolí.

V rámci segmentácie narastaním oblastí ide o snahu rozčleniť obraz na súvislé oblasti na základe kritéria homogenity danej určitou vlastnosťou (jasová úroveň, farba, textúra, tvar,...). V prípade postupu spájania oblastí je obraz prvotne rozdelený na veľké množstvo malých obrazových elementov, ktoré sa následne postupne spájajú do oblastí spĺňajúcich požadované kritérium. Opačný prístup je uplatňovaný pri štiepení oblastí, kedy je celistvá oblasť postupne rozdeľovaná do homogénnych celkov [5].

Optická lokácia objektov na oblohe predstavuje v podstate pozorovanie kontrastných elementov na viac-menej jednotnom pozadí. V tomto prípade by mohlo postačovať využitie jednoduchého globálneho prístupu prahovania pre zabezpečenie segmentácie záujmových letiacich objektov od pozadia a ich vyseparovania.

3. METÓDY DETEKČIE A KLASIFIKÁCIE POHYBUJÚCICH SA OBJEKTOV

V predchádzajúcej kapitole boli predstavené niektoré prístupy pre spracovanie a úpravu získaných individuálnych snímok s cieľom zefektívnenia ich analýzy z hľadiska porozumenia obsahu. V kontexte našej uvažovanej problematiky optickej detekcie a rozpoznávania letiacich objektov popísané metódy zabezpečia prípravu jednotlivých snímok už s určitou pravdepodobnosťou obsahujúcich záujmové letiace objekty, aby mohla byť následne realizovaná ich presná lokalizácia vyjadrením súradníc v obraze a ich klasifikácia.

Najskôr je však potrebné determinovať, ktorými snímkami je vhodné sa zaoberať a spracovávať ich, keďže v prípade pozorovania priestoru pomocou kamery, ktorému sa venuje táto práca, ide o analýzu dynamického obrazu, teda sekvencie snímok s malými časovými rozstupmi. Táto kapitola sa preto bude zaoberať možnosťami detekcie zmien v slede obrazov, ktoré indikujú pohyb v scéne snímanej statickou kamerou a umožnia určiť prítomnosť lietajúceho objektu v obraze.

Ďalej budú v rámci kapitoly prezentované postupy klasifikácie detegovaných pohybujúcich sa objektov, najmä letiacich, založené na umelej inteligencii a medzi nimi aj algoritmus vybraný pre praktickú realizáciu v nadväzujúcej bakalárskej práci.

3.1 Detekcia pohybu v sekvencii obrazov

Sekcia predstaví metódy detekcie pohyblivého objektu v slede snímok získaných statickou kamerou.

Optický tok

Optický tok popisuje zmeny v obraze za krátky časový interval dt . Metóda priradí každému bodu v obraze dvojrozmerný vektor rýchlosti (u,v) reprezentujúci veľkosť a smer pohybu daného pixelu.

Výpočet optického toku sa opiera o dva základné predpoklady: jasová hodnota každého bodu je v čase stála a pohyb bodov v blízkom okolí má podobný charakter. Pracuje s reprezentáciou dynamického obrazu ako funkcie $f(x,y,t)$, ktorá popisuje každý bod (x,y) v čase t . Na základe vyššie spomenutých predpokladov môžeme usúdiť, že pre bod posunutý o vzdialenosť (dx, dy) za časový interval dt platí:

$$f(x, y, t) = f(x + dx, y + dy, t + dt) \quad (3.1)$$

Pravú stranu vzťahu je možné aproximovať rozvinutím do Taylorovho radu a zanedbaním členov vyšších rádov pre veľmi malé hodnoty dx , dy , dt , získavame vyjadrenie rovnice optického toku:

$$-f_t = f_x \frac{dx}{dt} + f_y \frac{dy}{dt} \quad (3.2)$$

Vektor $(\frac{dx}{dt}, \frac{dy}{dt})$ reprezentuje vektor rýchlosti (u, v) . Určenie optického toku je však na základe vzťahu (3.2) extrémne náročné, využívajú sa tak pomocné výpočtové metódy ako „Lucas-Kanade“ či „Horn-Schunck“ [5], [9].

Diferenčná metóda

Diferenčná metóda vychádza z jednoduchého princípu odčítania dvoch snímok získaných v rôznych časových okamihoch.

Porovnávané snímky sú najskôr prevedené na šedo-tónové obrazy $f_1(x,y)$ a $f_2(x,y)$, na základe ktorých je determinovaný binárny rozdielový obraz $d(x,y)$:

$$d(x,y) = \begin{cases} 0, & |f_1(x,y) - f_2(x,y)| \leq \varepsilon \\ 1, & \text{ostatné prípady,} \end{cases} \quad (3.3)$$

kde malé kladné číslo ε označuje zvolený prah, pre elimináciu chybných detekcií šumu.

Efektivita algoritmu je podmienená použitím staticky upevnenej kamery, stabilnými svetelnými podmienkami, vhodne nastaveným prahom a dostatočným kontrastom medzi objektom a pozadím [5].

Metóda odčítania pozadia (Background Subtraction)

Základnou myšlienkou metódy je detekcia pohybu v obraze na základe porovnávania snímky s referenčnou snímkou a oddelenie pohyblivých objektov od nemenného pozadia. Referenčná snímka predstavuje tzv. model pozadia zobrazujúci statickú pozorovanú scénu bez prítomnosti pohyblivých objektov a môže byť priebežne aktualizovaný. Efektivita algoritmu klesá pri premenlivom osvetlení či oblačnosti alebo pri výskyte dynamických prvkov v pozadí, ktoré predstavujú napríklad listy rozhýbané vetrom, hmla alebo dym [10].

Podľa štúdie [11] existuje viacero techník realizácie metódy odčítania pozadia, pričom ich odlišnosti sú dané postupom získavania referenčnej snímky pozadia, kvalitou jeho modelovania a spôsobom vyjadrenia rozdielov. Avšak všetky vychádzajú z rovnakého základného princípu, popísaného nasledujúcim vzťahom:

$$\lambda_t(s) = \begin{cases} 1, & \text{pre } m(I_{s,t}B_s) > \tau \\ 0, & \text{ostatné prípady} \end{cases} \quad (3.4)$$

Kde λ_t figuruje ako príznak pohybu v čase t na pozícii v pohybovej maske zodpovedajúcej pixelu $s(x,y)$. m značí rozdiel hodnoty pixelu s medzi snímkou $I_{s,t}$ v čase t a modelom pozadia B_s . τ predstavuje nastavenú prahovú úroveň [11].

3.2 Prehľad možností rozpoznávania pohybujúcich sa objektov

Pohybujúce sa objekty v obraze zachytené vyššie spomínanými postupmi je následne možné identifikovať. Pre potreby klasifikácie detegovaných objektov sa efektívne uplatnia klasifikátory založené na strojovom učení a trénované na súbore snímok zobrazujúcich príklady falošnej a pozitívnej detekcie záujmových objektov. Zo snímok sú extrahované rôzne druhy príznakov predstavujúcich popis skúmaných objektov. Na základe nich dokážu optimálne naučené klasifikátory zaradiť dané objekty do vopred určených tried [1], [12].

V podkapitole budú popísané niektoré metódy uplatniteľné najmä v kontexte problematiky, na ktorú sa zameriava táto práca. Jednotlivé postupy sa odlišujú na základe použitého klasifikátora či podľa extrakcie príznakov a ich charakteru.

Rozšíreným klasifikačným algoritmom popísaným v štúdiu [13] je AdaBoost, ktorého princíp spočíva vo vytvorení silného klasifikátora, ktorý predstavuje váženú sumu viacerých slabých klasifikátorov. V tréningovej fáze sú opakovaným predkladaním vstupných vzoriek – príznakových vektorov – postupne adaptované váhy jednotlivých klasifikátorov podľa ich úspešnosti, a to s cieľom minimalizovať celkovú chybu klasifikácie [1].

V článku [1] zameriavajúcom sa na detekciu vtákov a ich rozlišovanie podľa druhu v okolí veternej elektrárne sú predstavené možnosti kombinácie algoritmu AdaBoost s tromi rôznymi typmi príznakov. Prvá najjednoduchšia možnosť vychádza z RGB hodnôt pixelov, teda klasifikácia prebieha na základe vybraného úseku farebného spektra.

Ďalším popisovaným prístupom je klasifikácia využívajúca Haarove príznaky, na ktorých je s využitím AdaBoost založený detekčný algoritmus prezentovaný v štúdiu autorov Viola – Jones [14], efektívny pri detekcii tváří. Haarove príznaky predstavujú odozvy na špeciálne čierno-biele filtre a odzrkadľujú kontrast v obraze. Ich hodnota je determinovaná rozdielom sumy intenzít pixelov spadajúcich do bielej oblasti filtra od sumy intenzít pixelov v tmavej oblasti [1].

Pri rozlišovaní letiacich objektov bol v článku [1] v kombinácii s klasifikátorom AdaBoost využitý aj deskriptor príznakov HOG (*Histogram of Oriented Gradients*) zabezpečujúci extrakciu informácií týkajúcich sa hrán a približných tvarov objektov na základe gradientov, ktorých veľkosť je v oblasti zobrazujúcej hrany najvýraznejšia. Príznaky mapujú rozloženie orientácií gradientov vo výseku obrazu [15]. Metóda bola prvýkrát predstavená pri detekcii ľudských postáv v obraze [16].

Rozpoznávanie pohybujúcich sa objektov popísaných pomocou príznakových vektorov môže byť zabezpečené aj SVM (*Support Vector Machines*) klasifikátorom. Algoritmus na základe označených tréningových dát determinuje optimálnu nadrovinu

(*optimal hyperplane*) separujúcu n -rozmerné prvky uvažovaných tried. Vzniknutá rovina by sa mala nachádzať v maximálnej nožnej vzdialenosti od tréningových vzorov, pričom vzory každej triedy umiestnené najbližšie sa nazývajú podporné vektory (*support vectors*). V prípade lineárne neseparovateľných dát je možné využiť tzv. jadrový trik (*kernel trick*), ktorý spočíva v transformácii dát pomocou vhodnej kernel funkcie do priestoru vyššej dimenzie, v ktorom ich separácia bude realizovateľná. Medzi často používané kernel funkcie patria polynomiálne (napr. kvadratické či kubické) alebo RBF kernel (*Radial Basis Function kernel*). Klasický SVM predstavuje binárny klasifikačný nástroj, rozlišujúci iba medzi dvoma triedami. Pre klasifikáciu do viacerých tried sa využíva kombinácia viacerých binárnych SVM, ktoré pracujú na princípe porovnávania vždy dvoch tried medzi sebou (*One Against One*) alebo jednej triedy voči všetkým ostatným (*One Against All*) [17], [18].

Metóda predstavená v literatúre [19] využíva SVM klasifikátor pri detekcii letiacich vtákov založenej na použití zjednodušenej reprezentácie vtácej kostry ako súbore príznakov. Základnou myšlienkou je podobnosť siluety všetkých letiacich vtákov, ktorá je zároveň dostatočne rozlíšiteľná od iných objektov. Popisovaný algoritmus tak dokáže určiť či zachytený objekt predstavuje letiaceho vtáka alebo nie.

Článok [1] ukazuje, že klasifikáciu pohybujúcich sa objektov zvládnu efektívne zabezpečiť aj konvolučné neurónové siete CNN, predstavujúce algoritmy hlbokého učenia. Namiesto spracovávania vopred získaných príznakových vektorov si zo skúmaných obrazov dôležité príznaky vyberajú samé pomocou predtrénovaných kernelov tvoriacich konvolučné vrstvy a realizujúcich konvolúciu so vstupným obrazom. Každá konvolučná vrstva je nasledovaná vrstvou pooling, ktorá zredukuje objem spracovávaných dát. Príznaky extrahované jednotlivými vrstvami CNN sú zoradené v plnerepojenej vrstve, ktorá na základe nich realizuje klasifikačnú úlohu a determinuje triedu vstupného obrazu [1].

3.3 Identifikácia objektov na základe významných bodov – SIFT

Podkapitola predstaví metódu, ktorá bola zvolená pre implementáciu pri spracovávaní videonahrávok za účelom detekcie a klasifikácie letiacich objektov v rámci tejto bakalárskej práce.

Metóda spočíva v extrakcii kľúčových príznakov zo spracovávaného obrazu pomocou algoritmu SIFT. Príznaky sú následne zapuzdrené do klastrov formujúcich príznakový vektor, ktorý predstavuje vstup pre SVM klasifikátor.

Metóda SIFT [20], [21] (*Scale-Invariant Feature Transform*) umožňuje detegovať alebo klasifikovať objekty pomocou extrakcie ich najvýraznejších lokálnych príznakov.

Vyznačuje sa nezávislosťou na mierke a rotácii, aj odolnosťou voči jasovým či geometrickým transformáciám. Tieto výhody ovplyvnili rozhodnutie vybrať danú metódu pre budúcu praktickú realizáciu. Tvorba príznakových vektorov je rozdelená do nasledujúcich etáp:

- 1) Detekcia extrémov vo vnútri scale-space
- 2) Lokalizácia významných bodov
- 3) Určenie orientácie významných bodov
- 4) Zostavenie deskriptoru významných bodov

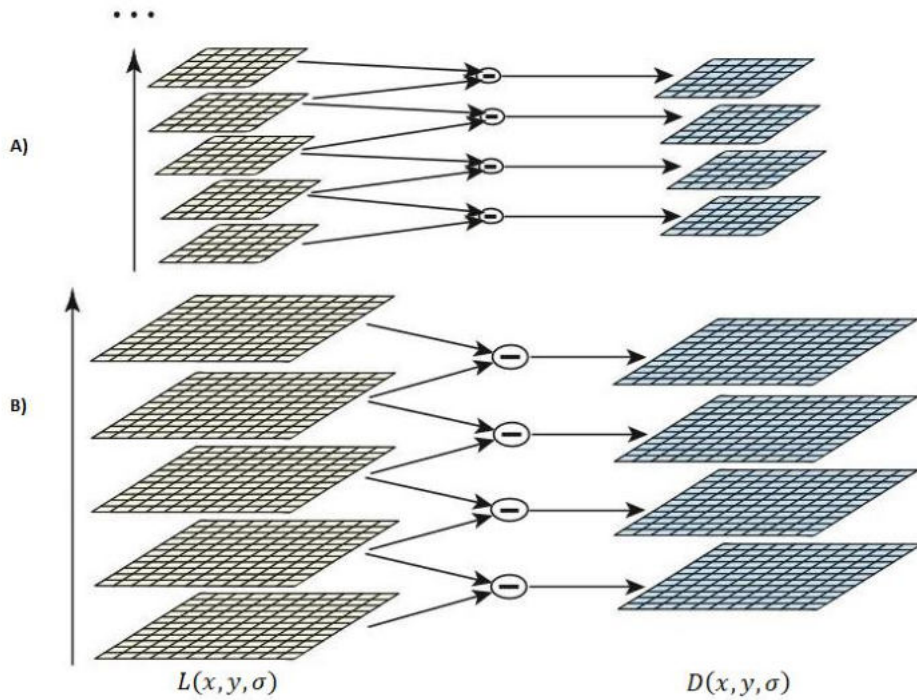
Zo vstupnej snímky je najskôr realizovaný mierkovo nezávislý priestor scale-space, ktorý predstavuje viacvrstvovú reprezentáciu obrazu. Prvá vrstva zodpovedá pôvodnému obrazu a každú ďalšiu tvorí obraz s vyššou mierkou. Scale-space sa člení na oktávy, pričom každá ďalšia reprezentuje obraz o polovičných rozmeroch oproti predchádzajúcej. Následne je využitím konvolúcie Gaussovho jadra $G(x, y, \sigma)$ so vstupným obrazom $I(x, y)$ pre rôzne šírky jadra σ , teda pre rôzne mierky, vytvorená tzv. Gaussova pyramída, zložená z obrazov $L(x, y, \sigma)$.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} \quad (3.5)$$

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (3.6)$$

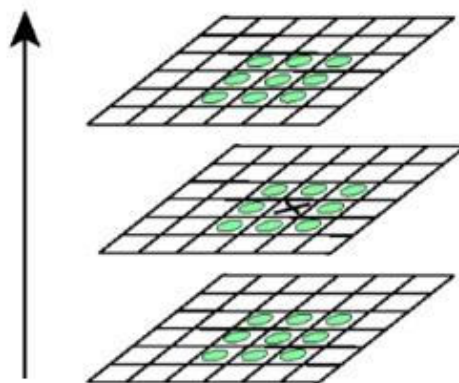
Medzi jednotlivými vrstvami pyramídy sú určované rozdielové obrazy $D(x, y, \sigma)$ podľa vzťahu, kde koeficient k indikuje zmenu mierky medzi susednými poschodiami:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3.7)$$



Obr. 3.1: Mierkovo nezávislá reprezentácia obrazu scale-space s konštrukciou rozdielových snímok $D(x,y,\sigma)$ [21]. A) 2.oktáva scale-space, B) 1.oktáva scale-space. Šípka značí zmenu mierky σ .

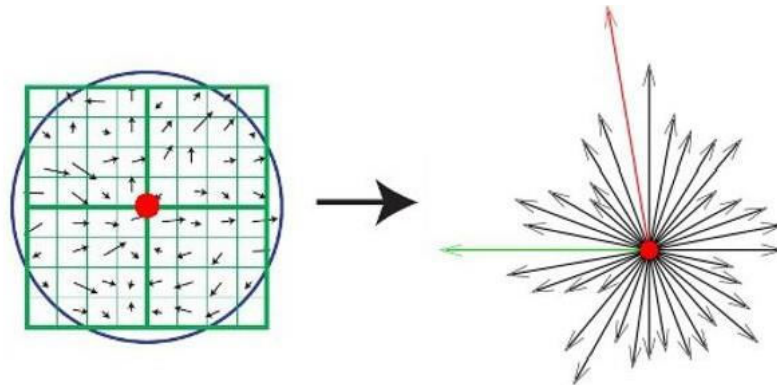
Porovnávaním každého pixelu s 26 susednými bodmi sú v rozdielových snímkach determinované lokálne extrémny, ktoré sa stávajú kandidátmi na významné body objektu.



Obr. 3.2: Vyhľadávanie lokálnych extrémov na základe porovnania bodu X s označenými bodmi v jeho okolí [21]. Šípka značí zmenu mierky σ .

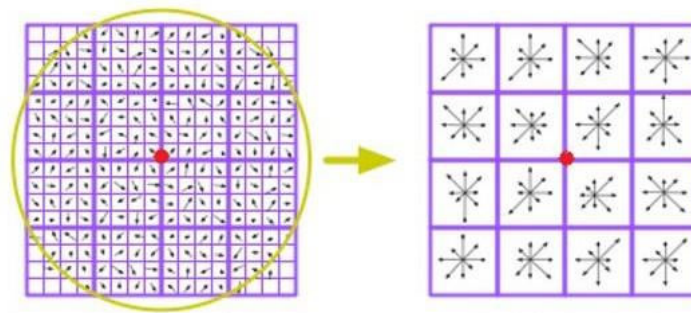
V ďalšej fáze pomôžu nadväzujúce výpočtové operácie určiť presnejšiu sub-pixelovú/sub-mierkovú polohu extrémov. Z množiny kandidátov sú navyše eliminované pixely s nízkym kontrastom a pixely nachádzajúce sa v blízkosti hrán, keďže sa vyznačujú nestabilitou, ktorá nie je pri kľúčových bodoch žiaduca. Výsledné významné body sú vďaka svojej presnej lokalizácii v scale-space nezávislé na mierke.

Tretia etapa spočíva v priradení dominantnej orientácie determinovaným bodom, čo zabezpečí ich invariantnosť voči rotácii. Na základe vypočítaných hodnôt gradientov z okolia kľúčového bodu je umožnená konštrukcia histogramu orientácií, ktorý vďaka 36 binom zahŕňa celý rozsah rotácie o 360° . Globálne maximum indikuje dominantnú orientáciu (červená farba na Obr. 3.3). Avšak pre významné lokálne maximá (aspoň 80% hodnoty globálneho) sú v rovnakých súradniciach vytvorené ďalšie významné body so zodpovedajúcou orientáciou (zelená farba na Obr. 3.3).



Obr. 3.3: Vypočítané gradienty v okolí významného bodu (vľavo) a histogram (vpravo) s dominantnými orientáciami (červená a zelená šípka) [21].

Podstatou posledného kroku je vytvorenie deskriptoru popisujúceho okolie významných bodov nezávisle od jasových aj geometrických transformácií. Okolie bodu je rozdelené na 16 menších oblastí, aby sa znížila citlivosť deskriptoru voči malým posunom obrazu. Výsledný 128-binový (16 x 8-binový histogram) deskriptor je tak zložený z histogramov zodpovedajúcich jednotlivým oblastiam okolia [21].



Obr. 3.4: Výsledný deskriptor významného bodu (vpravo) zostavený z gradientov v okolí (vľavo) [21].

Extrahované deskriptory však predstavujú veľmi široký súbor príznakov, ktorého veľkosť je navyše variabilná v závislosti od skúmaného objektu. V záujme vytvorenia efektívne pracujúceho klasifikačného algoritmu je vhodné mu predkladať rovnako dlhé vstupy, ideálne čo najmenšie [12]. Tejto požiadavke je možné vyhovieť využitím vhodnej metódy klastrovej analýzy, ktorá umožní zapuzdrenie príznakov do zvoleného počtu skupín – klastrov na základe určitých spoločných vlastností. Histogramy mapujúce výskyt klastrov v spracovávanom obraze tak sformujú príznakové vektory s dĺžkou rovnajúcou sa počtu uvažovaných klastrov predkladané klasifikátoru. Efektivita takéhoto postupu zoskupovania dát pred klasifikáciou bola dokázaná v štúdiu [22].

Skúmané objekty popísané sformovanými príznakovými vektormi budú môcť byť následne rozdeľované do zvolených tried, čo bude v rámci tejto bakalárskej práce úlohou SVM testovaného s využitím rôznych kernelových funkcií.

Predstavený postup je inšpirovaný metódou zo štúdie [12], ktorá sa zaoberala príbuznou tematikou (detekcia letiacich vtákov v prímorskom prostredí) a vykazovala veľmi uspokojivé výsledky. Obdobný prístup ku klasifikácii zahŕňajúci využitie SVM algoritmu a príznakov SIFT redukovaných do špeciálnych klastrov bol realizovaný aj v rámci štúdie [23] rozlišujúcej druhy vtákov.

4. DÁTA VYUŽÍVANÉ PRI VÝVOJI METÓDY

Videozáznamy spracovávané v rámci tejto práce boli získané pomocou kamery umiestnenej na strome v súkromnej záhrade neďaleko letiska Brno-Tuřany.

Zvolená kamera Xiaomi YI Outdoor 1080p je vhodná na použitie v exteriéri. Umožňuje prenos dát iba prostredníctvom wifi, takže bolo potrebné vytvoriť hotspot využitím Raspberry PI 3 B. K mikropočítaču v Raspberry s integrovaným wifi modulom vytvárajúcim wifi sieť v blízkom okolí, bol pripojený modul pre mobilné dáta s vloženou SIM kartou. Dáta sú zhromažďované na vysokokapacitnej SD karte vo vnútri kamery, z ktorej je dostupné online sťahovanie.

Použitá kamera však vykazuje nedostatky, pretože jej vstavaná funkcia „motion-detection“ nevyhovuje požiadavkám zamerania tejto práce. Zariadenie nie je dostatočne citlivé, aby bolo pri pohybe lietadiel, vtákov či ich krídlov na oblohe spustené nahrávanie. Z toho dôvodu boli zaznamenávané videosekvencie z celého dňa [24].

Získané dáta boli následne spracovávané prostredníctvom programu v jazyku C# na platforme .NET. Ich úprava pozostáva z niekoľkých krokov a je možné ju rozdeliť na spracovanie obrazových dát a spracovanie videozáznamov.

Spracovanie jednotlivých snímok

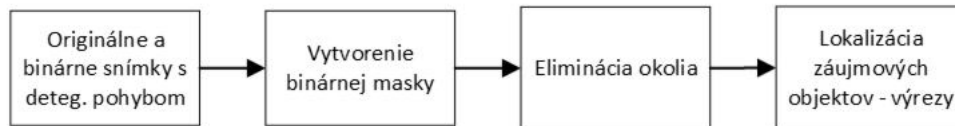
- 1) Načítanie obrazu a jeho transformácia do šedo-tónovej reprezentácie.
- 2) Prevod šedo-tónového obrazu na čiernobiely pomocou stanoveného prahu.
- 3) Využitie algoritmu [25] z knižnice ImageProcessor pre detekciu hrán v čiernobielym obraze.
- 4) Uloženie snímky.

Spracovanie videozáznamu

- 1) Načítanie videa a jeho rozdelenie na jednotlivé po sebe idúce sekvencie dlhé 1 s.
- 2) Aplikácia rovnakého postupu spracovania, aký bol popísaný vyššie pre prípad jednotlivých snímok, na každý obraz sekvencie.
- 3) Porovnávanie jednotlivých pixelov v dvoch nasledujúcich snímkach. Pixely, ktoré sa vyskytujú v oboch obrazoch na rovnakej pozícii a v čiernobielej reprezentácii sú indikované bielou farbou, budú v testovanom obraze zafarbené načierno.
- 4) Kontrola zostávajúcich bielych pixelov v testovanej snímke. Pri prekročení nastavenej hranice počtu ďalších bielych pixelov v okolí uvažovaného bielo bodu, je skúmaný obraz porovnávaný s predchádzajúcim.
- 5) V prípade, že sa biele pixely nenachádzajú aj v predošlej snímke, ide o správne zachytenie pohybujúceho sa prvku v obraze.
- 6) Uloženie obrazu s detegovaným pohybom.

5. URČOVANIE POLOHY OBJEKTU V OBRAZE

V tejto časti práce bude predstavený priebeh praktickej realizácie lokalizácie pohybujúcich sa objektov na oblohe v snímkach, ktoré boli predspracované využitím diferenčného algoritmu popísaného v predchádzajúcej kapitole.



Obr. 5.1: Postup lokalizácie letiaceho objektu v obraze.

5.1 Zisťovanie obrazových súradníc objektu

Uložené rozdielové snímky indikujúce pohyb v obraze a im zodpovedajúce originály boli analyzované s cieľom určenia súradníc pozície zachytených pohybujúcich sa objektov. Pre tento účel bol použitý program Matlab vo verzii R2021b. Podstatné boli jeho voliteľné súčasti Image Processing Toolbox a Computer Vision Toolbox. Všetky použité funkcie spomínané v texte práce sú popísané v dokumentácii [26].

Ku spracovávaným snímkam je pristupované ako k dvojrozmerným (prípadne viacrozmerným – RGB snímky) poliam, kde je každý pixel označený súradnicami (x,y) a charakterizovaný svojou hodnotou 0 až 255.

Vyseparovanie záujmových pohybujúcich sa objektov v binárnych reprezentáciách obrazu zabezpečené postupom uvedeným v predchádzajúcej kapitole nie je dokonalé, pretože algoritmus je citlivý aj na nežiaduce zmeny v pozadí vyvolané napríklad vetrom (vejúce listy stromov) či na pohyb, ktorý sa nevyskytol na oblohe. Skúmané binárne snímky je preto s cieľom eliminácie aktívnych pixelov neindikujúcich záujmové objekty na oblohe vhodné ďalej upravovať.

Na základe prvej originálnej snímky každej videosekvencie je najskôr vytvorená binárna maska segmentujúca obraz na skúmaný priestor oblohy zafarbený nabielo a na čierne okolie, ktoré pri analýze nesmie byť brané do úvahy. Využívaná je funkcia *superpixels()*, ktorá rozdelí vstupný obraz na zadaný počet regiónov. Z regiónov zodpovedajúcich okoliu mimo oblohy, ktoré je nutné filtrovať, je sformovaná tmavá časť masky, ktorej prislúchajú pixely s hodnotou 0. Zvyšok obrazu je označený nabielo hodnotou pixelov 1. Následne je aplikovaná erózia funkciou *imerode()*, ktorá zabezpečí mierne rozšírenie tmavého pásma.



Obr. 5.2: Príklad binárnej masky použitej pre elimináciu pohybových príznakov z okolia.

Ďalší krok spočíva v násobení rozdielových čiernobielych snímok nesúcich informáciu o zachytenom pohybe pripravenou maskou, čo umožní filtráciu pohybových príznakov, ktoré neboli zaznamenané na oblohe.

Príkazom *regionprops()* s atribútom '*BoundingBox*' sú determinované rozmery a súradnice najmenších možných obdĺžnikových ohraničení zahŕňajúcich vždy jeden celý detegovaný objekt v binárnych obrazoch. Pričom za objekty sú považované všetky spojené elementy tvorené pixelmi, ktorých hrany alebo rohy sa navzájom dotýkajú.

Obdĺžnikové rámy aj s príslušnými súradnicami zodpovedajúcimi jeho ľavému hornému rohu sú vykreslené do pôvodnej snímky, ku ktorej sa vzťahuje spracovávaný rozdielový obraz.



Obr. 5.3: Invertovaná binárna rozdielová snímka pred (vľavo hore) a po úpravách (vpravo hore), finálna lokalizácia letiaceho objektu v snímke z videosekvencie (dolu).

V prípade spracovania samostatných snímok mimo videosekvencie bol aplikovaný obdobný prístup, avšak binárna maska bola vytváraná osobitne pre každý spracovávaný obraz. Nepodarilo sa eliminovať falošnú detekciu, dokonca bola výraznejšia a algoritmus menej efektívny než v prípade videí. Dôvodom je, že v tomto prípade sa prítomnosť a poloha letiacich objektov na oblohe vyhodnocuje z obrazov so zvýraznenými hranami, čo predstavuje úplne iný prístup k riešenej problematike. Príklad aplikácie postupu zobrazuje Obr. 5.4.



Obr. 5.4: Invertované zobrazenie hrán pred (vľavo hore) a po úpravách (vpravo hore), finálna detekcia letiacich objektov v samostatnej snímke (dolu). Pre prehľadnosť nie sú zobrazené súradnice.

5.2 Zhodnotenie postupu

Testovaná metóda zvládne na základe čiernobielych rozdielových snímok z videosekvencie indikovať v obraze letiaci objekt na oblohe orámovaním a zobrazením súradníc pixelu zodpovedajúcemu jeho ľavému hornému rohu.

Nepodarilo sa však úplne zabrániť nežiaducej falošnej detekcii a v niektorých prípadoch vygenerovaná maska nedokáže pokryť celú oblasť mimo oblohy, takže vo výsledku sú okrem letiacich indikované aj iné pohybujúce sa objekty mimo priestor oblohy (oblačnosť, listy, konáre,...). Príklad nedostatočnej eliminácie pohybových príznakov mimo oblohy ilustruje Obr. 5.5.



Obr. 5.5: Invertovaná binárna rozdielová snímka pred (vľavo hore) a po úpravách (vpravo hore), finálna lokalizácia letiaceho objektu na oblohe aj pohybujúcich sa objektov v okolí, ktoré sa nepodarilo odfiltrovať maskou (dolu). Pre prehľadnosť nie sú zobrazené súradnice.

Problém falošnej detekcie iných ako letiacich objektov by mal vyriešiť klasifikátor založený na strojovom učení, ktorý by bol schopný takéto elementy zo všetkých detegovaných pohybujúcich sa objektov v obraze odfiltrovať. Návrh a priebeh testovania klasifikačného algoritmu budú popísané v nasledujúcej kapitole.

6. NÁVRH KLASIFIKAČNÉHO ALGORITMU

Kapitola mapuje vývoj klasifikačnej metódy využívajúcej SIFT deskriptory zo spracovávaných obrazov, ktorá bola zvolená pre účely tejto práce a bližšie popísaná v kapitole 3.3. Implementácia klasifikačného algoritmu bola realizovaná v programe Matlab vo verzii R2021b a všetky používané funkcie uvádzané v texte práce sú popísané v dokumentácii [26].

6.1 Dataset

Úloha klasifikácie detegovaných letiacich objektov v obraze si v prvom rade vyžaduje špecifikáciu tried, do ktorých budú uvažované objekty zaradované a následne zostavenie vhodnej tréningovej množiny snímok. Na základe analýzy dostupných materiálov bolo zvolené delenie zaznamenaných objektov na tri triedy:

- *Bird* – vtáky,
- *Plane* – lietadlá a helikoptéry,
- *Negative* – všetky ostatné objekty v spracovávanom obraze detegované ako pohybujúce a nespádajúce do prvých dvoch kategórií. Väčšinou ide o konáre, listy či výraznú kopovitú oblačnosť.

Súbor snímok určený na tréningovanie klasifikátora bol zostavený z výrezov z časti dostupných videozáznamov (viď 4. Dáta využívané pri vývoji metódy). Dané výrezy reprezentantov uvažovaných tried boli získané prostredníctvom lokalizačnej metódy popísanej v predchádzajúcej kapitole. Pre zabezpečenie uspokojujúcich výsledkov učenia však bolo nutné tréningovú množinu rozšíriť aj snímkami z ďalších zdrojov. Z fotografií krdľov letiacich na oblohe, boli využité samostatné výrezy jednotlivých vtákov. Množina bola ďalej doplnená aj snímkami z voľne dostupnej zbierky CIFAR-10, ktorá pozostáva z farebných snímok o veľkosti 32x32 pixelov delených do desiatich tried, medzi ktorými sú aj lietadlá a vtáky [27]. Pre potreby tejto práce však bolo možné využiť iba zopár vhodných príkladov lietadiel, pretože vtáky sú v rámci CIFAR-10 zobrazované v priveľkom detaile a v rôznych prostrediach. Nezodpovedajú reprezentáciám vtákov, s ktorými sa stretávame v analyzovaných videonahrávkach v rámci tejto práce.

Veľkosť snímok v zostavenej množine sa väčšinou pohybuje približne v rozmedzí od 2x2 pixelov (konáre, listy, oblaky, veľmi vzdialené objekty) až po 40x40 pixelov (detailnejšie reprezentácie lietadiel a vtákov), pričom prevažujú snímky menších rozmerov.

Zloženie vytvorenej množiny mapuje Tabuľka 6.1 a príklady zahrnutých snímok sú zobrazené na Obr. 6.1. Celková množina bola rozdelená na dáta určené na učenie klasifikátora a dáta pre jeho následné testovanie v pomere 90:10.

Tabuľka 6.1: Početnosť reprezentácií jednotlivých tried v rámci tréningovej množiny

Trieda	Zdroj			Spolu
	Videozáznamy	Fotografie	CIFAR-10	
Plane	675	18	1301	1994
Bird	84	1060	-	1144
Negative	2321	-	-	2321



Obr. 6.1: Ukážky obrázkov z množiny snímok pre učenie klasifikátora. V prvom riadku sú zobrazené reprezentácie lietadiel, v druhom vtákov.

6.2 Parametre pre evaluáciu modelov

Evaluácia a vzájomné porovnanie vytvorených modelov klasifikátorov budú v kontexte tejto práce realizované na základe niekoľkých štatistických parametrov [28], [29], definovaných pomocou matice zámen (viď Tabuľka 6.2).

Tabuľka 6.2: Matica zámen

		Predikovaná trieda	
		pozitívna	negatívna
Skutočná trieda	pozitívna	skutočne pozitívna (TP)	falošne negatívna (FN)
	negatívna	falošne pozitívna (FP)	skutočne negatívna (TN)

- 1) *Accuracy* vyjadruje celkovú úspešnosť klasifikácie skúmaného modelu na základe pomeru počtu správne klasifikovaných prípadov voči všetkým uvažovaným v rámci všetkých tried dohromady.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6.1)$$

Pri nerovnomernom zložení tréningovej množiny však tento parameter môže vykazovať zavádzajúce hodnoty, preto je nutné pri evaluácii brať do úvahy aj ďalšie.

- 2) *Precision (Positive Predictive Value PPV)* – presnosť – popisuje mieru správnej predikcie výskytu triedy, teda aká časť z pozitívne predikovaných prípadov bola pozitívna aj v skutočnosti.

$$Precision = \frac{TP}{TP + FP} \quad (6.2)$$

- 3) *Recall (True Positive Rate TPR)* – senzitivita – vypovedá o schopnosti modelu správne odhaliť skutočne pozitívne prípady a predstavuje pomer pozitívnej predikcie danej triedy ku všetkým jej pozitívnym prípadom výskytu.

$$Recall = \frac{TP}{TP + FN} \quad (6.3)$$

- 4) *Skóre F1* popisuje vzťah medzi predchádzajúcimi dvoma parametrami a reprezentuje ich harmonický priemer.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (6.4)$$

- 5) *ROC krivka* zobrazuje závislosť medzi mierou falošne pozitívnych prípadov v rámci všetkých negatívnych (*False Positive Rate FPR*) a senzitivitou (*TPR*), ktorá už bola definovaná vyššie.

$$FPR = \frac{FP}{FP + TN} \quad (6.5)$$

Dôležitým parametrom pre hodnotenie modelu klasifikátora je plocha pod ROC krivkou označovaná ako AUC (*Area Under the Curve*). Vypovedá o schopnosti modelu správne rozlišovať medzi pozitívnymi a negatívnymi prípadmi skúmanej triedy. Čím vyššia je hodnota AUC, tým uspokojivejšie sú výsledky klasifikácie.

6.3 Priebeh učenia klasifikátora

Zozbierané obrazové dáta tvoriace tréningovú množinu boli najskôr označené menovkou príslušnej triedy a medzi sebou premiešané.

Vstupné farebné snímky sú prevádzané na šedotónové, aby na ne bolo možné aplikovať nasledujúce funkcie. Pomocou funkcie *extractFeatures()* s atribútom označujúcim voľbu SIFT metódy sú z každého obrazu vypočítané SIFT deskripty, reprezentované 128-prvkovými vektormi. Vzhľadom ku skutočnosti, že spracovávané snímky sú väčšinou veľmi malé, bolo nutné určovať príznaky SIFT z celého obrazu, nie len z jeho významných bodov, čo bolo testované najskôr. Avšak v prípade takého postupu nebolo z väčšiny spracovávaných snímok možné získať dostatok informácií, teda deskriptorov, potrebných pre popis objektov a rozlišovanie medzi jednotlivými triedami.

Všetky SIFT deskripty získané analýzou obrazov tvoriacich množinu určenú na natrénovanie klasifikátora, boli spracované iteratívnym algoritmom *k-means*, realizovaným funkciou *kmeans()*, čo predstavuje prvú fázu učenia navrhovaného klasifikačného algoritmu. Ide o postup rozdelenia vstupných dát do vopred určeného počtu k klastrov, definovaných súradnicami svojich stredov C . Každý bod je zaradený do klastra, ku ktorého stred sa nachádza najbližšie, pričom je vzdialenosť počítaná na základe Euklidovskej metriky [30]. Výstupom použitej funkcie je okrem lokalizácie stredov C vzniknutých klastrov aj vektor priradujúci každému SIFT deskriptoru jediná hodnotu v rozsahu 1 až k , teda označenie klastra, do ktorého je príslušný deskriptor zahrnutý.

V nasledujúcom kroku sú zostavené histogramy mapujúce početnosť klastrov v jednotlivých spracovávaných obrázkoch, ktoré predstavujú popisné vektory s jednotnou dĺžkou k .

Takéto príznakové popisy snímok v tréningovej množine sú ďalej využité v druhom kroku učenia algoritmu. Slúžia na trénovanie SVM na rozlišovanie medzi triedami *bird*, *plane*, *negative*.

V rámci testovania boli využitím nástroja Matlab Classification Learner App [31] vytvárané modely SVM pracujúce s popisnými vektormi o rôznej dĺžke k , danej počtom klastrov, do ktorých boli delené SIFT príznaky. Testované boli modely využívajúce základnú metódu lineárnej separácie dát, ale aj rozdielne kernelové funkcie (kvadratická, kubická, RBF). Pre zabezpečenie relevantných výsledkov a zabránenie prílišnej adaptácii klasifikátora na tréningové dáta (*overfitting*) bola pri testovaní využitá krížová validácia (*cross-validation*). Spočívala v rozdelení vstupnej množiny príznakových vektorov na 5 podmnožín, z ktorých bola jedna vyčlenená na testovanie modelu natrénovaného na zvyšných štyroch. V rámci opakovaní procesu dostala vždy iná množina status testovacej a následne bola skúmaná priemerná chybovosť [32].

S cieľom evaluácie a porovnania boli pre každý trénovaný model počítané hodnoty parametrov definovaných v kapitole 6.2. Dosiahnuté výsledky sumarizuje Tabuľka 6.3. Pre všetky uvažované štatistické parametre platí, že čím vyššiu hodnotu dosahujú, tým sú výsledky uspokojivejšie. Na základe nich možno usúdiť, že testované modely indikujú veľkú úspešnosť klasifikácie. Hodnoty parametrov vo všetkých prípadoch prekonávajú hranicu 80% , ale často až 90% (úroveň skóre F1 nad 0.80, často nad 0.90). Najslabšie výsledky klasifikácie odzrkadľované parametrami TPR aj PPV sú vykazované v súvislosti s triedou vtákov (*Bird*) a môžu byť dôsledkom menej výrazného zastúpenia danej triedy v rámci tréningovej množiny. Dĺžka popisných vektorov k daná počtom skupín, do ktorých boli vopred kategorizované SIFT príznaky, nemá na úspešnosť modelov príliš výrazný vplyv, hodnoty pozorovaných parametrov sa pre jednotlivé veľkosti k zásadne nelíšia. Avšak aj tak možno označiť za všeobecne najuspokojivejšie modely pracujúce s príznakovými vektormi o dĺžke $k = 100$. Vychádzajúc z hodnôt pozorovaných parametrov bol pre každú hodnotu k vybraný a vyznačený (viď Tabuľka 6.3) najúspešnejší model SVM. Diagram na Obr. 6.2 zhŕňa celú predstavenú klasifikačnú metódu.

Tabuľka 6.3: Úspešnosť klasifikátorov spracúvajúcich SIFT deskriptory

k	Kernel funkcia	Accuracy [%]	Trieda								
			Plane			Bird			Negative		
			TPR [%]	PPV [%]	F1	TPR [%]	PPV [%]	F1	TPR [%]	PPV [%]	F1
50	Lineárna	92,4	91,2	89,4	0,903	81,0	85,4	0,831	89,4	98,3	0,936
	Kvadratická	94,7	93,1	92,7	0,929	89,3	88,3	0,888	98,8	98,5	0,986
	Kubická	94,4	93,0	92,9	0,929	88,1	88,3	0,882	98,8	98,8	0,988
	RBF	94,8	92,6	94,0	0,933	90,7	87,8	0,892	98,6	99,1	0,988
100	Lineárna	93,7	91,5	92,0	0,917	87,4	85,5	0,864	98,7	99,3	0,990
	Kvadratická	94,7	93,4	93,5	0,934	90,1	88,5	0,893	98,8	98,9	0,988
	Kubická	95,0	93,4	93,7	0,935	89,7	88,9	0,893	98,9	99,0	0,989
	RBF	94,8	92,1	94,2	0,931	91,7	86,6	0,891	98,6	99,6	0,991
200	Lineárna	93,4	91,5	91,4	0,914	86,1	85,5	0,858	98,6	99,0	0,988
	Kvadratická	94,5	93,3	92,2	0,927	87,4	88,7	0,880	98,6	98,9	0,987
	Kubická	94,5	92,8	93,1	0,929	88,8	88,0	0,884	98,7	98,9	0,988
	RBF	94,1	91,8	92,9	0,923	88,6	86,2	0,874	98,7	99,0	0,988



Obr. 6.2: Postup navrhutej metódy klasifikácie na základe SIFT deskriptorov

6.4 Posilnenie príznakových vektorov – informácie o farbe

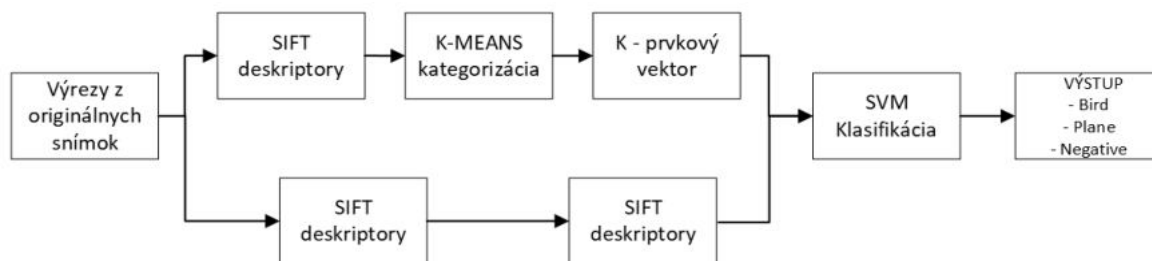
Práca sa síce zameriava na analýzu farebných snímkov, avšak používaná metóda SIFT je aplikovateľná iba na ich šedotónové reprezentácie, takže v doposiaľ navrhnutých modeloch SVM nebola farba braná do úvahy. Aj farebné príznaky však môžu byť dôležitým parametrom pre rozlíšenie jednotlivých tried, ktorý by mohol zvýšiť úspešnosť klasifikácie. Napríklad v článku [1] bola pre rozpoznávanie letiacich vtákov implementovaná okrem iných aj metóda využívajúca príznaky z RGB farebného priestoru, a to s veľmi uspokojivými výsledkami.

Ďalšia fáza testovania v rámci tejto bakalárskej práce sa teda zameriava na skúmanie úspešnosti klasifikácie SVM modelov pracujúcich s príznakovými vektormi rozšírenými o farebnú informáciu.

Pre každý spracovávaný obraz bol vypočítaný histogram z RGB priestoru, pozostávajúci z 8 binov pre všetky tri farebné kanály (*Red, Green, Blue*). Následne sa z jeho 8x8x8 prvkov predstavujúcich početnosť jednotlivých kombinácií farebných kanálov sformoval vektor o dĺžke 512 elementov, ktorý predĺžil vektor zostavený na základe SIFT deskriptorov.

Modely SVM boli ďalej trénované pri rovnakých nastaveniach ako v prvej fáze testovania, avšak počet klastrov pre kategorizáciu SIFT deskriptorov bol ustálený na hodnote 100.

Úspešnosť naučených modelov bola opäť hodnotená na základe štatistických parametrov definovaných v kapitole 6.2 a ich hodnoty zobrazuje Tabuľka 6.4, kde je model SVM vykazujúci najlepšie hodnoty vyznačený. Možno skonštatovať, že použitím RGB príznakov sa podarilo mierne zlepšiť dosiahnutú úspešnosť klasifikácie a to najmä v prípade triedy *Bird*, kde pozorované parametre už väčšinou tiež presahujú hodnotu 90% (v prípade F1 úroveň 0.90). Postup modifikovanej metódy klasifikácie sumarizuje diagram na Obr. 6.3.



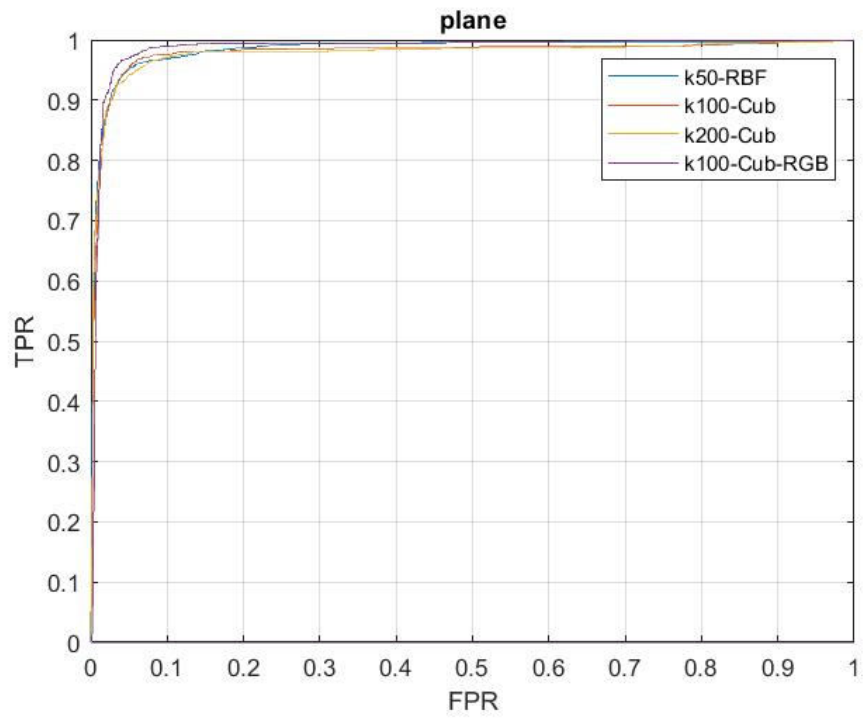
Obr. 6.3: Priebeh klasifikácie realizovanej na základe SIFT deskriptorov aj RGB histogramov

Tabuľka 6.4: Úspešnosť klasifikátorov spracúvajúcich popisy snímok tvorené SIFT deskriptormi aj RGB histogramami

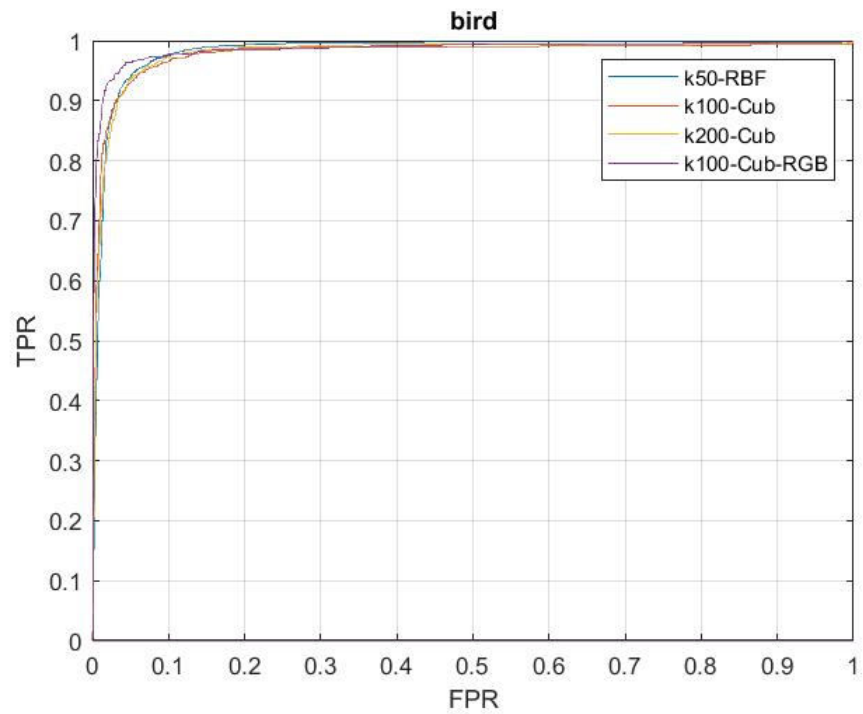
			Trieda								
			Plane			Bird			Negative		
k	Kernel funkcia	Accuracy [%]	TRP [%]	PPV [%]	F1	TRP [%]	PPV [%]	F1	TRP [%]	PPV [%]	F1
100	Lineárna	95,2	94,1	92,9	0,935	89,3	91,0	0,901	98,9	99,1	0,990
	Kvadratická	95,9	94,8	94,4	0,946	91,9	91,9	0,919	98,8	99,2	0,990
	Kubická	96,2	95,7	94,2	0,949	91,7	93,7	0,927	98,8	99,2	0,990
	RBF	93,7	94,0	89,6	0,917	85,1	90,6	0,878	97,8	99,0	0,984

6.5 Vzájomné porovnanie výsledných modelov SVM

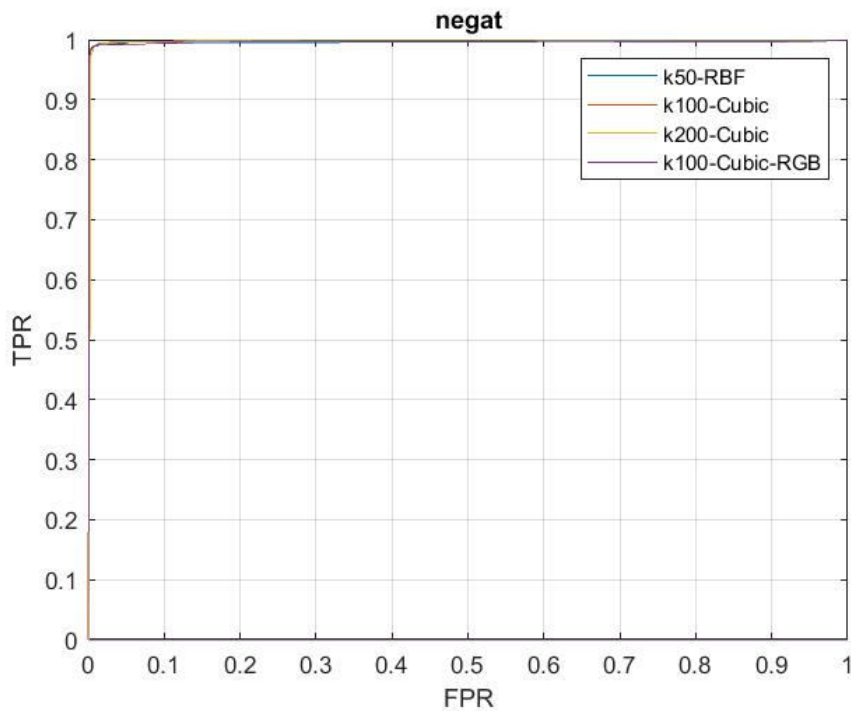
Tabuľka 6.3 a Tabuľka 6.4 obsahujú farebne vyznačené modely SVM boli medzi sebou ďalej porovnávané prostredníctvom ROC kriviek, a to z hľadiska úspešnosti klasifikácie triedy *Plane* na Obr. 6.4, triedy *Bird* na Obr. 6.5 a triedy *Negative* na Obr. 6.6. Tabuľka 6.5 uvádza veľkosti plôch pod zobrazenými ROC krivkami.



Obr. 6.4: ROC pre triedu lietadiel (*Plane*)



Obr. 6.5: ROC pre triedu vtákov (*Bird*)



Obr. 6.6: ROC pre triedu *Negative*

Tabuľka 6.5: Vypočítané hodnoty AUC pre vybrané naučené modely SVM

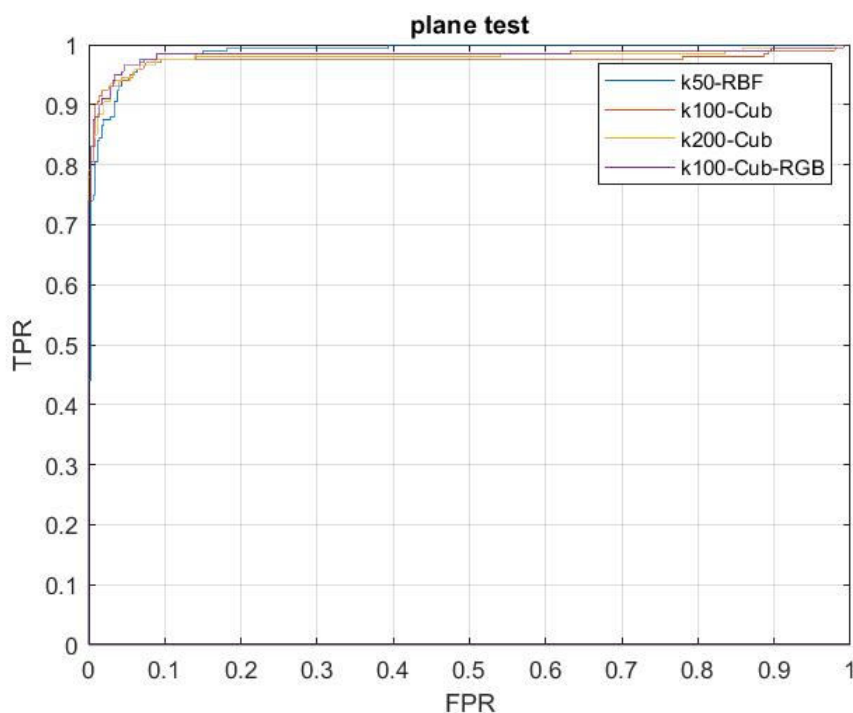
Príznamy	k	Legenda v ROC grafoch	Kernel funkcia	AUC train		
				Plane	Bird	Negative
SIFT	50	K50-RBF	RBF	0,98	0,98	1,00
SIFT	100	K100-Cubic	Kubická	0,98	0,98	1,00
SIFT	200	K200-Cubic	Kubická	0,98	0,98	1,00
SIFT + RGB	100	K100-Cubic-RBF	Kubická	0,99	0,99	1,00

Grafy a hodnoty AUC ukazujú, že úspešnosť uvažovaných modelov je porovnateľná. Avšak za model s najuspokojivejšími výsledkami možno považovať SVM s kubickou kernelovou funkciou realizujúci rozlišovanie tried na základe SIFT deskriptorov doplnených o RGB histogramy (v legendách označený ako k100-Cub-RGB). V oboch zobrazených grafoch je jeho ROC krivka najstrmšia, najviac sa blíži k ľavému hornému rohu a aj plocha AUC ňou definovaná je najväčšia.

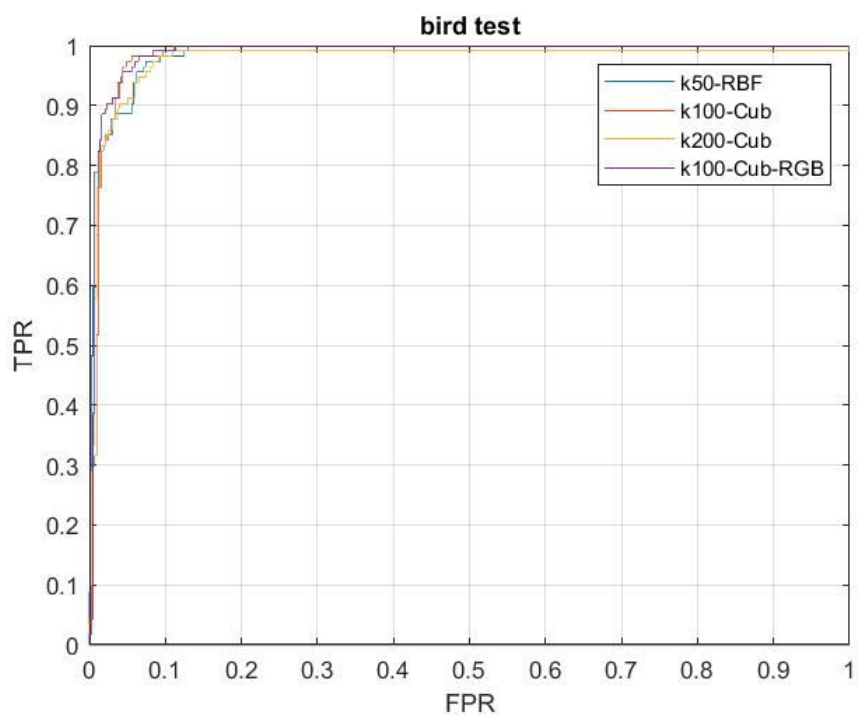
Následne bolo pristúpené k overovaniu úspešnosti uvedených štyroch modelov využitím 10-tich % z celkovej množiny, ktoré boli už na začiatku vyčlenené ako dataset pre testovanie. Opäť boli určované hodnoty štatistických parametrov, ktoré mapuje Tabuľka 6.6. Výsledky boli porovnávané aj z hľadiska vývoja ROC kriviek pre jednotlivé triedy, vyobrazených na Obr. 6.7, Obr. 6.9 a Obr. 6.8. Tabuľka 6.7 prezentuje k nim prislúchajúce hodnoty plôch AUC.

Tabuľka 6.6: Overenie výsledkov vybraných SVM modelov na testovacej množine

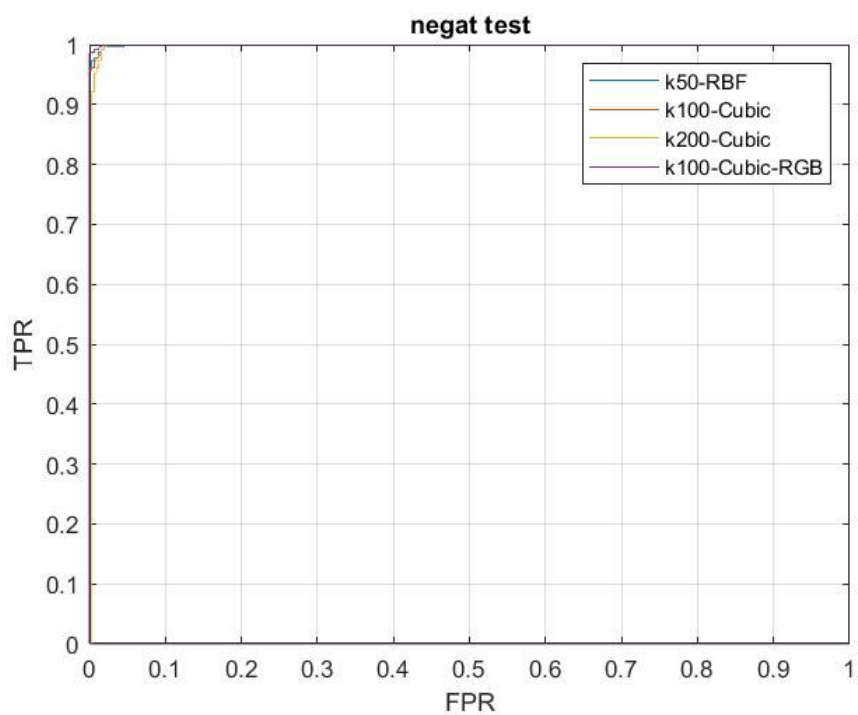
Príznamy	k	Kernel funkcia	Accuracy [%]	Trieda								
				Plane			Bird			Negative		
				TRP [%]	PPV [%]	F1	TRP [%]	PPV [%]	F1	TRP [%]	PPV [%]	F1
SIFT	50	RBF	94,0	92,3	93,6	0,929	90,4	86,9	0,886	98,7	99,4	0,990
SIFT	100	Kubická	94,5	93,5	92,9	0,932	88,3	89,1	0,887	98,8	98,9	0,988
SIFT	200	Kubická	94,1	93,5	94,2	0,938	88,6	86,3	0,874	97,4	97,8	0,976
SIFT + RGB	100	Kubická	94,9	95,5	94,8	0,951	92,0	87,4	0,896	98,6	98,7	0,986



Obr. 6.7: ROC pre triedu lietadiel (Plane) - testovacia množina



Obr. 6.9: ROC pre triedu vtákov (Bird) - testovacia množina



Obr. 6.8: ROC pre triedu *Negative* - testovacia množina

Tabuľka 6.7: Vypočítané hodnoty AUC – overovanie úspešnosti vybraných SVM modelov na testovacej množine

Príznamy	k	Legenda v ROC grafoch	Kernel funkcia	AUC test		
				Plane	Bird	Negative
SIFT	50	K50-RBF	RBF	0,99	0,99	1,00
SIFT	100	K100-Cubic	Kubická	0,97	0,99	1,00
SIFT	200	K200-Cubic	Kubická	0,98	0,98	1,00
SIFT + RGB	100	K100-Cubic-RBF	Kubická	0,98	0,99	1,00

Dosiahnutá úspešnosť všetkých modelov pri testovaní na nových vzorových snímkach potvrdila predchádzajúce veľmi uspokojivé predpoklady získané po naučení. Výsledky klasifikácie snímkov z testovacej množiny prostredníctvom štyroch overovaných modelov SVM sú na základe sledovaných štatistických parametrov aj priebehov ROC kriviek veľmi dobré a vzájomne porovnateľné, žiaden z modelov sa úspešnosťou nevyčleňuje od zvyšných. Využitie RGB príznakov tak v tomto prípade nenaznačuje výraznejší pozitívny či negatívny vplyv na schopnosť korektného rozlišovania tried.

7. TESTOVANIE NAVRHNUTÝCH METÓD DETEKČIE A KLASIFIKÁCIE

Navrhnuté čiastkové postupy prezentované v predchádzajúcich kapitolách mohli byť napokon spojené v komplexný algoritmus, ktorý bol podrobený praktickému testovaniu. Diagram na Obr. 7.1 sumarizuje jeho jednotlivé kroky. V nasnímaných minútových videozáznamoch oblohy, delených na samostatné snímky s odstupom jednej sekundy, boli najskôr prostredníctvom diferenčného algoritmu, porovnávajúceho binárne hranové reprezentácie snímok danej sekvencie, zachytené pohybové príznaky. Binárna maska tvorená z prvej snímky každej videonahrávky zabezpečila odfiltrovanie pohybových príznakov zaznamenaných mimo oblohu. Zvyšné príznaky umožnili lokalizáciu záujmových letiacich objektov. Tieto objekty boli extrahované formou obdĺžnikových výrezov z pôvodnej snímky. Využitím metód predstavených v predchádzajúcej kapitole boli vytvorené popisné vektory výrezov založené na SIFT deskriptoroch, prípadne doplnených aj o informácie z RGB histogramov. Popisy boli jednotlivo predkladané naučenému modelu SVM, ktorý na základe nich zaradil každý objekt do jednej z uvažovaných tried. Elementy klasifikované do tried *Bird* a *Plane* boli nakoniec v originálnych snímkach zvýraznené červeným orámovaním, obrazovými súradnicami a menovkou triedy. Prvky triedy *Negative* však označované neboli.



Obr. 7.1: Postup analýzy videozáznamov

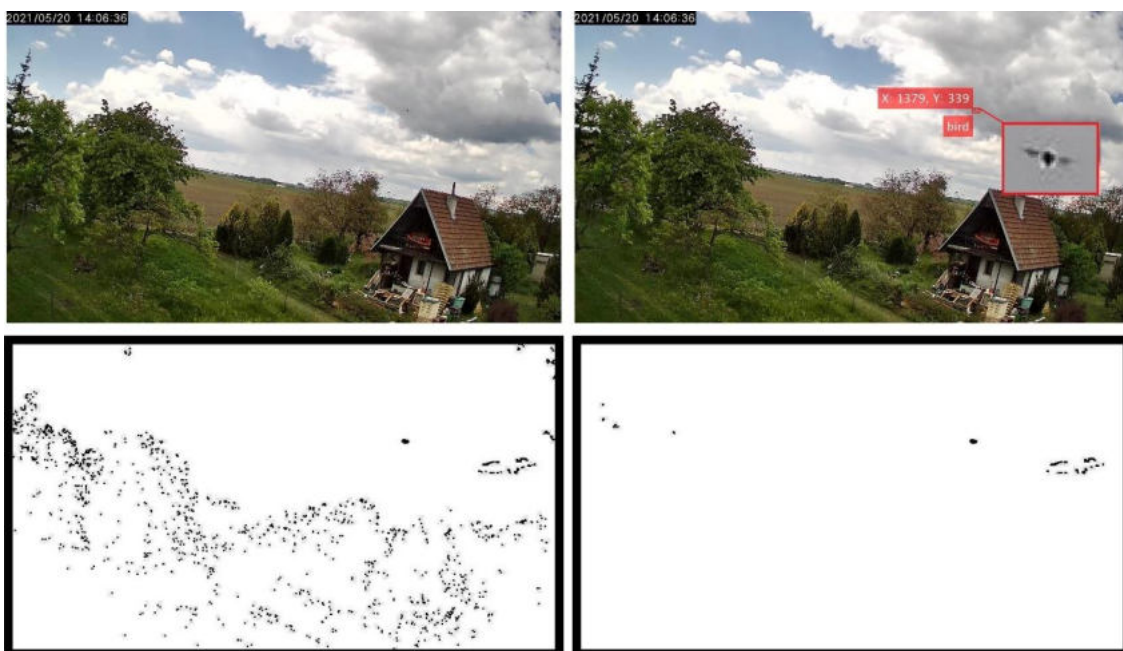
7.1 Výstupy analýzy videozáznamov

Testovanie vo viacerých prípadoch potvrdilo funkčnosť navrhovaného algoritmu a v spracovávaných videozáznamoch sa podarilo korektne lokalizovať, rozpoznať a označiť letiace objekty na oblohe. Príklady úspešnej detekcie sú prezentované na Obr. 7.2 a Obr. 7.3.

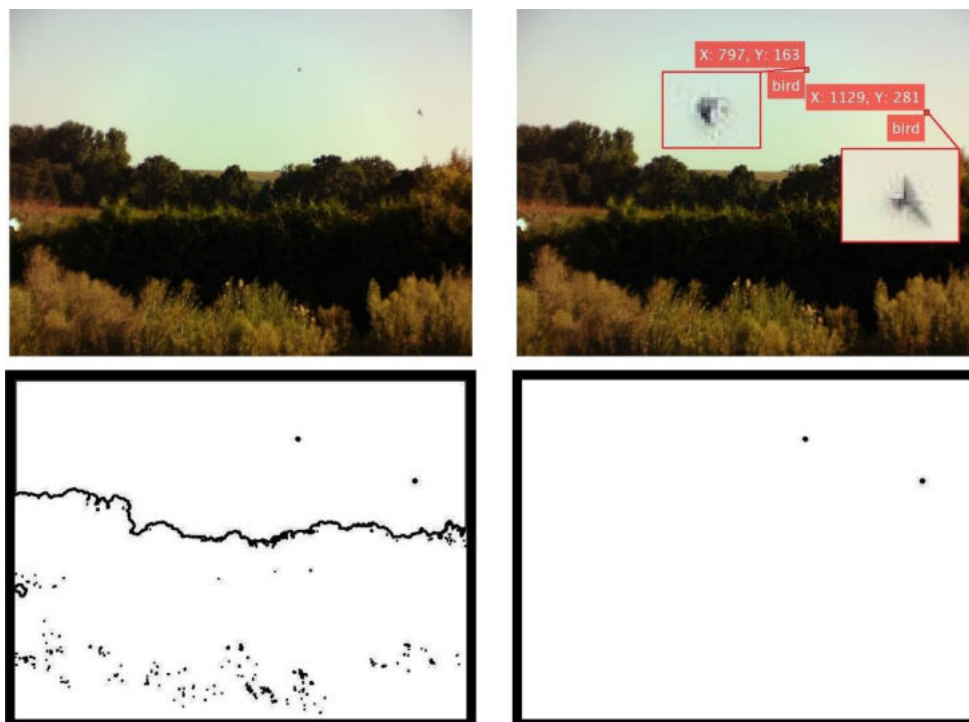
Algoritmus bol úspešne testovaný opäť aj na samostatných snímkach, ktoré nie sú súčasťou videosekvencie. Využitý bol obdobný postup ako v prípade videí, avšak pre lokalizáciu letiacich objektov v obraze boli namiesto pohybových príznakov v diferenčných snímkach použité binárne hranové reprezentácie. Príklad výsledku takejto analýzy fotografie je predstavený na Obr. 7.4



Obr. 7.2: Správna detekcia lietadla (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).



Obr. 7.3: Správna detekcia vtáka (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).



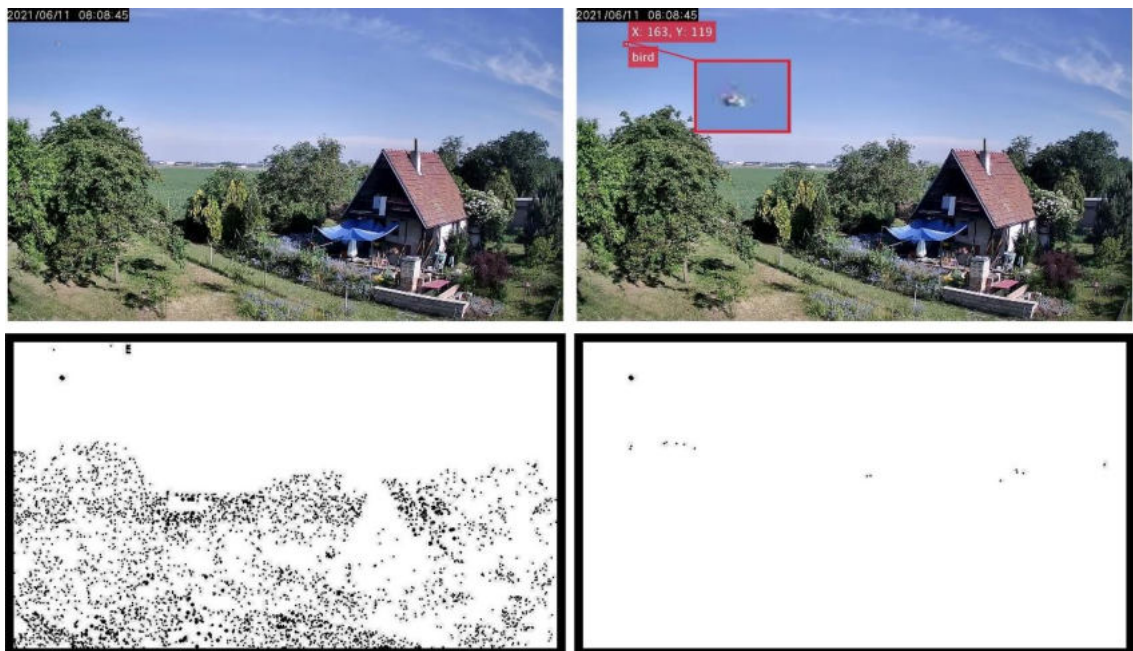
Obr. 7.4: Príklad spracovania samostatnej fotografie mimo videosekvencie - korektná detekcia vtákov (vpravo hore), originálna snímka (vľavo hore), pôvodná binárna hranová reprezentácia (vľavo dole), maskou filtrovaná hranová snímka (vpravo dole).

Klasifikačný algoritmus sa však v priebehu analýzy dopustil aj viacnásobných chybných predikcií. Vyskytli sa prípady zámien medzi triedami *Bird* a *Plane*, pričom v o niečo vyššej miere bolo pozorované nesprávne označovanie vtáka za lietadlo. Takýto príklad je ilustrovaný na Obr. 7.5. Opačný prípad mylnej klasifikácie lietadla ako vtáka zobrazuje Obr. 7.6.

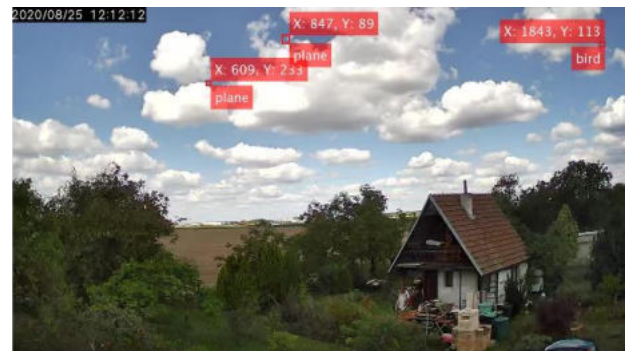
Najčastejšie prípady nesprávnej klasifikácie však predstavovali zaradenie reprezentantov triedy *Negative* medzi vtáky a lietadlá. Takéto prípady sa vyskytovali najmä pri zhoršených poveternostných podmienkach ako silný vietor, ktorý spôsobil výrazné zmeny v okolí, teda pohyb listov a konárov, a tým nedostatočnú filtráciu pohybových príznakov pri vyžití binárnej masky. Klasifikátor tak musel spracovať väčšie množstvo výrezov a stúpala pravdepodobnosť chybnnej predikcie. Ďalším negatívnym poveternostným javom je výrazná kopovitá oblačnosť, ktorá je často mylne označovaná za vtáky, avšak najmä za lietadlá. V prípade niektorých videí bol tento typ falošnej detekcie veľmi výrazný, čo indikuje nedostatočné zastúpenie príkladov oblačnosti v rámci tréningovej množiny, takže modely SVM ju nezvládli správne identifikovať. Spomenuté príklady nekorektnej klasifikácie sú zobrazené na Obr. 7.7.



Obr. 7.5: Nesprávne rozpoznanie vtáka ako lietadla (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).



Obr. 7.6: Chybné rozpoznanie lietadla ako vtáka (vpravo hore), originálna snímka (vľavo hore), pôvodná diferenčná snímka s pohybovými príznakmi (vľavo dole), maskou filtrovaná diferenčná snímka (vpravo dole).



Obr. 7.7: Pohybujúce sa objekty náležiacie triede *Negative* (vľavo listy, vpravo oblačnosť) nesprávne označené ako lietadlá a vtáky.

7.2 Časová analýza

Fungovanie algoritmu bolo pozorované aj z hľadiska časovej náročnosti. Osem minútových videí zobrazujúcich scénu pri variabilných poveternostných podmienkach, delených vždy na 60 samostatných snímok, bolo analyzovaných pomocou navrhutej metódy (vid' Obr. 7.1). Najskôr pri použití modelu SVM pracujúcom s popismi tvorenými iba na základe SIFT deskriptorov a následne pri opakovaní analýzy rovnakých videí bol využitý model rozlišujúci popisy doplnené aj o farebné príznaky. Pre oba prípady bola zaznamenaná doba potrebná pre vykonanie jednotlivých čiastkových operácií v prípade každej snímky sekvencie so zaznamenaným pohybom. Tabuľka 7.1 zobrazuje spracované výsledky.

Výsledky indikujú, že analýza snímky, ktorá zobrazuje pohybujúci sa prvok si vyžaduje približne tri sekundy, pričom časovo najnákladnejšia je počiatočná detekcia pohybových príznakov diferenčným algoritmom. Čas potrebný na spracovanie je premenlivý vzhľadom na počet lokalizovaných pohybujúcich sa prvkov, ktorý determinuje počet opakovaní operácií tvorby príznakových popisov, klasifikácie a prípadného označovania výsledkov. Postup využívajúci pre popis obrazov iba SIFT deskriptory bez farebných príznakov je podľa očakávaní menej časovo náročný, keďže pracuje s menším počtom krokov a s nižším objemom dát.

Tabuľka 7.1: Potrebná doba spracovania jednej snímky z videosekvencie, na ktorej bol zaznamenaný pohyb

Operácia		Doba spracovania			
		Priemer [ms]	Medián [ms]	Min [ms]	Max [ms]
Detekcia pohybu diferenciou snímok		3121	3068	3051	3492
Násobenie maskou + lokalizácia záujmových objektov		9,3	8,4	5,5	31,6
Príznakový popis	SIFT	17,4	15,9	10,7	36,8
	SIFT+RGB	66,8	59,1	45,2	190,4
Klasifikácia	SIFT	187,9	135,5	2,7	827,1
	SIFT+RGB	200,1	145,9	6,1	834,6
Identifikácia v obraze	SIFT	15,9	9,6	0,6	125,7
	SIFT+RGB	76,7	30,3	0,6	391,7
Spolu	SIFT	3351,3	3237,4	3070,3	4512,7
	SIFT+RGB	3473,7	3311,7	3108,2	4939,8

7.3 Zhodnotenie výsledkov

Navrhnutý postup detekcie a klasifikácie letiacich objektov na oblohe bol overovaný spracovaním dostupných videonahrávok a dosiahnuté výsledky je následne možné zbilancovať.

Diferenčný algoritmus detegujúci pohyb v sekvencii snímok sa ukázal ako dostatočne citlivý aj na vzdialenejšie letiace objekty, avšak pri výbere kamery s vyšším rozlíšením či pri jej vhodnejšom umiestnení (bližšie k snímanej scéne) by sa jeho účinnosť mohla zvýšiť. Nevýhodou postupu je pomerne veľká časová náročnosť, citlivosť na kopovitú oblačnosť a najmä zaznamenávanie pohybových príznakov aj mimo záujmový priestor oblohy, ktoré je však efektívne korigované v ďalších krokoch spracovania.

Využitím binárnej masky, tvorenej z prvej snímky každej sekvencie, segmentujúcej obraz na oblohu a okolie sa podstatnú časť pohybových informácií z priestoru mimo oblohy darí odfiltrovať. Pri silnom vetre v snímanej scéne sa však okolie rýchlo mení, menej tak korešponduje s maskou a viac nežiaducich pohybových príznakov (listy a konáre) ostáva neeliminovaných.

Na základe zostávajúcich pohybových príznakov sa lokalizačnému algoritmu spoľahlivo darí vyseparovať pohybujúce sa objekty z originálnej snímky vo forme samostatných výrezov, aby z nich mohli byť vyextrahované informácie potrebné pre popis a následnú klasifikáciu.

Výsledky klasifikácie možno označiť za uspokojivé, a to najmä pri optimálnych podmienkach v snímanej scéne, akými sú jasné počasie s nevýraznou oblačnosťou a bez

silného vetra. Vo viacerých prípadoch bola dokázaná schopnosť správneho rozpoznávania letiacich objektov. Avšak vyskytli sa aj prípady zámen medzi triedami vtákov a lietadiel ako aj nesprávne zaradenie prvkov triedy *Negative* do jednej z dvoch zvyšných tried. Pri zhoršených poveternostných podmienkach sú nároky na klasifikátor zvýšené kvôli už spomínaným limitom použitej binárnej masky a nutnosti spracovania väčšieho množstva výrezov zobrazujúcich prvky z okolia (listy a konáre). Pri výskyte kopovitej oblačnosti sa účinnosť klasifikátora veľmi znižuje. Limity by sa dali korigovať doplnením tréningovej množiny pre učenie SVM modelov o viac reprezentantov všetkých uvažovaných tried, najmä o príklady oblačnosti. Ďalším riešením zvyšujúcim efektivitu klasifikačného nástroja by mohlo byť využitie kamery s väčším rozlíšením, prípadne jej umiestnenie bližšie k snímanému priestoru. Bolo by tak možné zo zaznamenaných objektov extrahovať viac informácií potrebných pre klasifikáciu.

V rámci analýzy boli testované dva prístupy k popisu snímok pred klasifikáciou pomocou SVM: 1) využitie iba klastrovaných SIFT deskriptorov a 2) doplnenie SIFT deskriptorov o RGB histogramy. Hoci výsledky naučenia SVM modelov boli v prípade modelov doplnených o farebné príznaky o niečo uspokojivejšie (viď kapitola 6. Návrh klasifikačného algoritmu), pri spracovaní záznamov bola pri využití tohto prístupu na popis vo vyššej miere prítomná falošná detekcia lietadiel a vtákov v prípadoch, keď išlo o elementy triedy *Negative*. Porovnanie dvoch postupov tvorby popisu je zobrazené na Obr. 7.8. Modely využívajúce iba SIFT deskriptory tak možno označiť za efektívnejšie, a to aj z hľadiska nižšej časovej náročnosti.



Obr. 7.8: Porovnanie modelu klasifikátora využívajúceho iba SIFT deskriptory (vľavo) s modelom klasifikujúcim na základe SIFT aj RGB príznakov (vpravo).

ZÁVER

Bakalárska práca sa najskôr zamerala na teoretický rozbor problematiky pasívnej optickej detekcie objektov pohybujúcich sa na oblohe v dynamickom obraze. Boli predstavené základné možnosti úprav obrazu, ktoré zefektívňujú proces jeho následnej obsahovej analýzy a porozumenia. Ďalej sa práca venovala predstaveniu postupov umožňujúcich detekciu pohybových príznakov v po sebe nasledujúcich snímkach, ktoré determinujú prítomnosť pohybujúcich sa objektov v snímanom obraze. Bol zostavený prehľad algoritmov využívajúcich prostriedky umelej inteligencie pre zabezpečenie klasifikácie zachytených pohybujúcich sa objektov, ktoré boli použité v štúdiách s príbuzným zameraním ako táto práca.

Na základe zozbieraných informácií bol zostavený návrh komplexného algoritmu schopného v spracovávaných videonahrávkach letiaci objekt na oblohe detegovať, prostredníctvom obrazových súradníc lokalizovať a napokon rozhodnúť, či ide o vtáka, lietadlo alebo element patriaci do súhrnnej skupiny zaznamenaných prvkov mimo záujmu (konáre, listy, oblačnosť, hmyz).

Algoritmus bol implementovaný primárne v programe Matlab, využívajúc však komponent zabezpečujúci detekciu pohybových príznakov diferenciou po sebe idúcich snímkov videosekvencie, ktorá bola realizovaná na platforme .NET v jazyku C#. Dôvodom je vyššia kvalita detekcie pohybu, než pri obdobnej implementácii v programe Matlab.

Úloha klasifikácie zaznamenaného objektu, realizovaná prostredníctvom SVM je založená na popise obrazu s využitím SIFT deskriptorov, prípadne aj jeho doplnením o farebné príznaky vo forme histogramov z RGB priestoru. Testovaním bolo zistené, že prístup využívajúci samotnú SIFT metódu bez informácií ohľadom farby je efektívnejší.

Posledná časť tejto práce sa venovala overovaniu funkčnosti celého detekčného programu analýzou dostupných videozáznamov vzdušného priestoru. Výsledky potvrdili schopnosť algoritmu úspešne zaznamenať, rozpoznať a označiť letiaci objekt na oblohe. Odhalili však aj jeho nedostatky v podobe viacnásobných chybných predikcií pri klasifikácii a najmä jeho citlivosť na zhoršené poveternostné podmienky, pri ktorých klesá spoľahlivosť korektnej detekcie a najmä rozpoznávania. Riešenie by mohlo spočívať v rozšírení tréningovej množiny o viac reprezentantov každej z tried a hlavne v jej doplnení o príklady kopovitej oblačnosti. Trénovaný SVM model by tak bol lepšie pripravený na rozlišovanie medzi uvažovanými triedami. Problematickým sa navyše ukázalo aj zaznamenávanie a klasifikácia veľmi vzdialených objektov, a to z dôvodu malého objemu informácií, ktoré sa k takýmto prvkom v obraze viažu. Tento nedostatok by mohol byť korigovaný využitím kvalitnejšej kamery s vyšším rozlíšením a prípadne jej vhodnejším umiestnením vzhľadom k snímanej scéne.

Zefektívnenie klasifikácie by mohlo byť dosiahnuté aj doplnením popisu spracovaných obrazov o ďalšie informácie. Napríklad o iný typ farebných príznakov, než

aké boli testované v rámci bakalárskej práce. Ďalšou možnosťou by mohlo byť rozšírenie schopností nástroja aj o spracovávanie audio informácií zo snímanej scény, ktoré by boli tiež zahrnuté do celkového rozhodovania o klasifikácii zaznamenaného objektu.

Po vhodných úpravách a individuálnych adaptáciách vzhľadom ku zvolenej praktickej aplikácii by mohol navrhnutý algoritmus nájsť uplatnenie napríklad v systémoch zameraných na ochranu sádov, viníc a súkromných záhrad, využívajúcich mechanizmy plašenia voči detegovaným vtákom škodiacim na úrode.

LITERATÚRA

- [1] YOSHIHASHI, R., R. KAWAKAMI, M. IIDA a T. NAEMURA. Bird detection and species classification with time-lapse images around a wind farm: Dataset construction and evaluation. *Wind Energy* [online]. 2017, 20(12), 1983-1995 [cit. 2021-12-19]. ISSN 10954244. Dostupné z: doi:10.1002/we.2135
- [2] HLADÍK, Roman. *Pasivní optická detekce pohybujících se cílů* [online]. Brno, 2013 [cit. 2021-12-19]. Dostupné z: <http://hdl.handle.net/11012/24875>. Diplomová práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky. Vedoucí práce doc. Ing. Pavel Fiala, Ph.D.
- [3] TREIBER, Marco Alexander. *An Introduction to Object Recognition* [online]. 1. London: Springer London, 2010 [cit. 2021-12-19]. Advances in Pattern Recognition. ISBN 978-1-84996-234-6. Dostupné z: doi:10.1007/978-1-84996-235-3
- [4] NEŠPOR, Zdeněk. *Sledování objektu ve videosekvenci* [online]. Brno, 2013 [cit. 2021-12-19]. Dostupné z: <http://hdl.handle.net/11012/26549>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Vedoucí práce Ing. Petr Číka, Ph.D.
- [5] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. *Image processing, analysis, and machine vision*. 4th edition. United States of America: Cengage Learning, 2015. ISBN 978-1-133-59369-0.
- [6] BHUYAN, Manas Kamal. *Computer Vision and Image Processing* [online]. 1. Boca Raton, FL: CRC Press/Taylor & Francis Group: CRC Press, 2019 [cit. 2021-12-19]. ISBN 9781351248396. Dostupné z: doi:10.1201/9781351248396
- [7] YANG, Kecheng. Depth Estimation Cameras, Pinhole Geometry, and Stereo: Pinhole camera (4/48). *SlidePlayer* [online]. [cit. 2021-12-19]. Dostupné z: <https://slideplayer.com/slide/12452941>
- [8] JANÁKOVÁ, Ilona. *Úvod a motivace*. Brno, 2021. Dostupné také z: 01_Uvod a motivace.pdf (vutbr.cz). Prednáška z predmetu MPC-POV (Computer Vision / Počítačové vidění).
- [9] NOVOTNÝ, Josef. *Detekční algoritmy pohybujících se objektů* [online]. Brno, 2019 [cit. 2021-12-19]. Dostupné z: <http://hdl.handle.net/11012/81737>. Bakalářské práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Vedoucí práce Ing. Petr Marcoň, Ph.D.
- [10] KARASULU, Bahadır a Serdar KORUKOGLU. *Performance Evaluation Software* [online]. 1. New York, NY: Springer New York, 2013 [cit. 2021-12-19]. SpringerBriefs in Computer Science. ISBN 978-1-4614-6533-1. Dostupné z: doi:10.1007/978-1-4614-6534-8

- [11] BENEZETH, Y., P.M. JODOIN, B. EMILE, H. LAURENT a C. ROSENBERGER. Review and Evaluation of Commonly-Implemented Background Subtraction Algorithms. In: *2008 19th International Conference on Pattern Recognition* [online]. IEEE, 2008, s. 1-4 [cit. 2021-12-19]. ISBN 978-1-4244-2174-9. Dostupné z: doi:10.1109/ICPR.2008.4760998
- [12] T'JAMPENS, Roeland, Francisco HERNANDEZ, Florian VANDECASTEELE a Steven VERSTOCKT. Automatic detection, tracking and counting of birds in marine video content. In: *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)* [online]. IEEE, 2016, s. 1-6 [cit. 2021-12-19]. ISBN 978-1-4673-8910-5. Dostupné z: doi:10.1109/IPTA.2016.7821031
- [13] FREUND, Yoav a Robert E SCHAPIRE. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences* [online]. 1997, 55(1), 119-139 [cit. 2021-12-19]. ISSN 00220000. Dostupné z: doi:10.1006/jcss.1997.1504
- [14] VIOLA, P. a M. JONES. Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* [online]. IEEE Comput. Soc, 2001, I-511-I-518 [cit. 2021-12-19]. ISBN 0-7695-1272-0. ISSN 1063-6919. Dostupné z: doi:10.1109/CVPR.2001.990517
- [15] MALLICK, Satya. Histogram of Oriented Gradients explained using OpenCV. *LearnOpenCV* [online]. 2016 [cit. 2021-12-19]. Dostupné z: <https://learnopencv.com/histogram-of-oriented-gradients>
- [16] 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. 2021-12-12]. ISBN 0-7695-2372-2. Dostupné z: doi:10.1109/CVPR.2005.177
- [17] YU, Hwanjo a Sungchul KIM. SVM Tutorial — Classification, Regression and Ranking. ROZENBERG, Grzegorz, Thomas BÄCK a Joost N. KOK, ed. *Handbook of Natural Computing* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, 2012, s. 479-506 [cit. 2022-04-30]. ISBN 978-3-540-92909-3. Dostupné z: doi:10.1007/978-3-540-92910-9_15
- [18] YANG, Jiawei, Zeping WU, Ke PENG, Patrick N. OKOLO, Weihua ZHANG, Hailong ZHAO a Jingbo SUN. Parameter selection of Gaussian kernel SVM based on local density of training set. *Inverse Problems in Science and Engineering* [online]. 2021, 29(4), 536-548 [cit. 2022-04-30]. ISSN 1741-5977. Dostupné z: doi:10.1080/17415977.2020.1797716
- [19] QUNYU XU a XIAOFENG SHI. A simplified bird skeleton based flying bird detection. In: *Proceeding of the 11th World Congress on Intelligent Control and Automation* [online]. IEEE, 2014, s. 1075-1078 [cit. 2021-12-19]. ISBN 978-1-4799-5825-2. Dostupné z: doi:10.1109/WCICA.2014.7052867

- [20] LOWE, D.G. Object recognition from local scale-invariant features. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision* [online]. IEEE, 1999, 1150-1157 vol.2 [cit. 2021-12-19]. ISBN 0-7695-0164-8. Dostupné z: doi:10.1109/ICCV.1999.790410
- [21] HRÚZ, Marek. *SIFT, SURF, MSER* [online]. Plzeň, 2015. Dostupné také z: <http://www.kky.zcu.cz/uploads/courses/mpv/04/materialy04.pdf>
- [22] CSURKA, G., C.R. DANCE, L. FAN, J. WILLAMOWSKI a C. BRAY. Visual categorization with bags of keypoints. In: *Workshop on Statistical Learning in Computer Vision, ECCV* [online]. Prague, Czech Republic, May 2004, s. 1-22 [cit. 2022-04-30]. Dostupné z: https://www.researchgate.net/publication/228602850_Visual_categorization_with_bags_of_keypoints
- [23] MARINI, A., A. J. TURATTI, A. S. BRITTO a A. L. KOERICH. Visual and acoustic identification of bird species. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* [online]. IEEE, 2015, s. 2309-2313 [cit. 2021-12-19]. ISBN 978-1-4673-6997-8. Dostupné z: doi:10.1109/ICASSP.2015.7178383
- [24] ŠIRŮČKOVÁ, Anna. *Odborná zpráva o postupu prací a dosažených výsledcích za rok 2020* [online]. Brno, 2020. Dostupné také z: https://docs.google.com/document/d/15t_yb0-Qn_F9L66ka3DuodGqV0zp6dOP/edit
- [25] DetectEdges. *ImageProcessor* [online]. [cit. 2021-12-19]. Dostupné z: <https://imageprocessor.org/imageprocessor/imagefactory/detectedges/>
- [26] Documentation. *MathWorks* [online]. [cit. 2022-04-30]. Dostupné z: https://www.mathworks.com/help/index.html?s_tid=CRUX_lftnav
- [27] CIFAR-10 and CIFAR-100 datasets. *Department of Computer Science, University of Toronto* [online]. [cit. 2022-04-30]. Dostupné z: <https://www.cs.toronto.edu/~kriz/cifar.html>
- [28] MINAEE, Shervin. 20 Popular Machine Learning Metrics. Part 1: Classification & Regression Evaluation Metrics. In: *Towards Data Science* [online]. 2019 [cit. 2022-05-07]. Dostupné z: <https://towardsdatascience.com/20-popular-machine-learning-metrics-part-1-classification-regression-evaluation-metrics-1ca3e282a2ce>
- [29] NARKHEDE, Sarang. Understanding AUC - ROC Curve. In: *Towards Data Science* [online]. 2018 [cit. 2022-05-07]. Dostupné z: <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
- [30] K-means clustering - MATLAB kmeans. *MathWorks* [online]. [cit. 2022-04-30]. Dostupné z: <https://www.mathworks.com/help/stats/kmeans.html#namevaluepairarguments>
- [31] Classification Learner App - MATLAB & Simulink. *MathWorks* [online]. [cit. 2022-04-30]. Dostupné z: https://www.mathworks.com/help/stats/classification-learner-app.html?s_tid=CRUX_lftnav

- [32] Select Data for Classification or Open Saved App Session - MATLAB & Simulink. *MathWorks* [online]. [cit. 2022-04-30]. Dostupné z: <https://www.mathworks.com/help/stats/select-data-and-validation-for-classification-problem.html>

ZOZNAM SKRATIEK

AUC	Area Under the Curve
CIFAR-10	Canadian Institute for Advanced Research, 10 classes
CNN	Convolutional Neural Network
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GPS	Global Positioning System
HOG	Histogram of Oriented Gradients
PPV	Positive Predictive Value
RBF	Radial Basis Function
RGB	Red, Green, Blue
ROC	Receiver operating characteristic
SD	Secure Digital
SIFT	Scale-Invariant Feature Transform
SIM	Subscriber Identity Module
SVM	Support Vector Machines
TN	True Negative
TP	True Positive
TPR	True Positive Rate

ZOZNAM PRÍLOH

PRÍLOHA A – ZDROJOVÉ KÓDY A UKÁŽKY DÁT	61
---	-----------

Príloha A – Zdrojové kódy a ukážky dát

Príloha je v elektronickej podobe uložená na USB disku a pozostáva zo zdrojových kódov programu a príkladov spracovávaných dát, rozdelených do nasledujúcich priečinkov:

1. Dataset
2. Príprava učenia
3. Spracovanie videí 1
4. Spracovanie videí 2
5. Spracovanie fotografií 1
6. Spracovanie fotografií 2

Ďalej USB disk zahŕňa aj súbor *Komentáre.docx*, ktorý obsahuje popis jednotlivých priečinkov.

Úspešné spustenie zdrojových kódov je možné iba na zariadení s nainštalovaným programom Matlab (verzia R2021b alebo novšia). Podstatné sú aj balíčky Image Processing Toolbox a Computer Vision Toolbox.