



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

DEPARTMENT OF INTELLIGENT SYSTEMS

PROTIDRONOVÁ OCHRANA PERIMETRU

ANTI-DRONE PERIMETER PROTECTION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ROMAN JANÍK

VEDOUcí PRÁCE

SUPERVISOR

prof. Ing. MARTIN DRAHANSKÝ, Ph.D.

BRNO 2019

Zadání bakalářské práce



21617

Student: **Janík Roman**
Program: Informační technologie
Název: **Protidronová ochrana perimetru**
Anti-Drone Perimeter Protection
Kategorie: Zpracování obrazu

Zadání:

1. Prostudujte současné metody a nástroje pro detekci objektů ve videu. Vytvořte si přehled o současných přístupech k problematice detekce rychle se pohybujících a obtížně rozeznatelných objektů.
2. Navrhnete systém pro detekci a lokalizaci dronu či hejna dronů s ohledem na ochranu perimetru.
3. Vámi navržený systém z předchozího bodu implementujte.
4. Otestujte systém se skutečnými drony v terénu.
5. Porovnejte dosažené výsledky a diskutujte možnosti budoucího vývoje.

Literatura:

- Popovic V. et al. *Near-infrared high-resolution real-time omnidirectional imaging platform for drone detection*. In: Target and Background Signatures II, International Society for Optics and Photonics, 2016.
- Bandyopadhyay S. *Object of interest detection in video sequence using co-segmentation: A new era in video surveillance*. In: 4th International Conference on Recent Advances in Information Technology (RAIT), IEEE, 2018. s. 6.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

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

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

Datum zadání: 1. listopadu 2018

Datum odevzdání: 15. května 2019

Datum schválení: 1. listopadu 2018

Abstrakt

Rozvoj technologie dronů s sebou přináší příležitosti pro mnoho oblastí lidské činnosti, ale zároveň i bezpečnostní hrozby. Vzniká potřeba těmto hrozbám efektivně čelit. V této práci je popsána problematika a současné metody pro detekci objektů ve videu zachyceném pohybující se kamerou. Dále jsem navrhl systém pro detekci a lokalizaci dronu či hejna dronů. Algoritmus pro detekci je založen na konvoluční neuronové síti, konkrétně na algoritmu SSD. Systém jsem implementoval pomocí knihovny OpenCV s možnou akcelerací algoritmu na GPU pomocí OpenCL. Vytvořené řešení jsem otestoval jak na videu, tak na video výstupu kamery.

Abstract

Development of drone technology brings opportunities for many fields of human activity, but simultaneously brings security threats. A need to effectively face these threats arises. In this work, problematics and state-of-the-art methods for object detection in a video captured by moving camera. Further I proposed a system for a drone or a flock of drones detection and localization. Algorithm for detection is based on convolutional neural network, specifically on SSD algorithm. I implemented the system with library OpenCV with possible acceleration of algorithm with GPU via OpenCL. I tested the created solution on both a video and a video camera output.

Klíčová slova

detekce objektů, konvoluční neuronové sítě, SSD, TensorFlow, dron, bezpilotní letoun, video dohled, OpenCV, bezpečnostní hrozby, pohybující se kamera

Keywords

object detection, convolutional neural networks, SSD, TensorFlow, drone, UAV, video surveillance, OpenCV, security threats, moving camera

Citace

JANÍK, Roman. *Protidronová ochrana perimetru*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Ing. Martin Drahanský, Ph.D.

Protidronová ochrana perimetru

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana prof. Ing., Dipl.-Ing., Martina Drahanského, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Roman Janík
15. května 2019

Poděkování

V této sekci je možno uvést poděkování vedoucímu práce a těm, kteří poskytli odbornou pomoc (externí zadavatel, konzultant, apod.).

Obsah

1	Úvod	2
2	Přehled současného stavu	4
2.1	Reprezentace obrazu	4
2.2	Předzpracování obrazu	5
2.3	Detekce objektů	6
2.3.1	Metody založené na vzhledu	6
2.3.2	Metody založené na pohybu	7
2.3.3	Hybridní metody	8
2.4	Sledování objektů	8
2.5	Problémy spojené s detekcí pohybujících se objektů	9
2.6	Metriky pro vyhodnocení detektorů	13
2.7	Klasifikační algoritmy	15
2.8	Konvoluční neuronové sítě	15
2.9	Segmentace	15
2.9.1	Sémantická segmentace založená na Konvolučních neuronových sítích	16
2.9.2	Texture-Aware Superpixel Segmentation	17
2.10	Kosegmentace	17
2.10.1	Metody založené na Markovských náhodných polích	18
2.10.2	Color reward strategy and Active Contour Model	20
3	Návrh systému	24
3.1	Použitý software	24
3.2	Návrh aplikace	24
3.3	Uživatelské rozhraní	25
4	Implementace systému	26
5	Testování	27
6	Závěr	28
	Literatura	29

Kapitola 1

Úvod

První snahy o umožnění počítačovým systémům vidět, tedy dát jim schopnost zpracovat a interpretovat informace z obrazových dat, byly vyvinuty již v 60. letech 20. století. Od té doby byl zaznamenán značný pokrok v počítačovém vidění, jak se oblast zabývající se těmito problémy nazývá. Navzdory stáří je počítačové vidění velmi aktuální a rozvíjející se oblast počítačové vědy. V současnosti prostupuje oblast počítačového vidění do mnoha technických oborů jako kontrola dopravy, medicína, kontrola kvality výrobků, vojenství, robotika a inteligentní dohledové systémy. Poslední tři příklady se týkají této práce.

Spolu s počítačovým viděním se v posledních letech i rapidně rozvíjely drony neboli bezpilotní letouny. V této práci je jako dron považován pouze bezpilotní letoun, i když je možné se setkat s použitím označení „dron“ pro jiné bezpilotní robotické zařízení jako loď, či ponorka. Dron je řízen na dálku člověkem, nebo létá autonomně řízen palubním počítačem. Součástí vybavení dronu jsou různé senzory, např. pro měření vzdálenosti, gyroskopy, akcelerometry, kamery atd. To umožňuje využití v mnoha oblastech lidské činnosti, například ve vojenství, zemědělství, fotografování ze vzduchu, kontrole dopravy a dohledových systémech. Drony dnes přispívají k rozvoji těchto oblastí a se zvyšující se dostupností těchto systémů běžné veřejnosti také roste jejich počet ve vzdušném prostoru.

Díky možnosti volně létat, umožňují drony nebývale mnoho způsobů zneužití této technologie. Výčet těchto bezpečnostních hrozeb zahrnuje narušení soukromí, narušení letového provozu, špionáž, pašování kontrabandu, zbraní a drog, zakládání požárů, teroristické útoky a atentáty. Drony mohou nést nebezpečné výbušniny, chemikálie, radioaktivní nebo biologické zbraně. Mnoho z výše uvedených hrozeb se již stalo skutečností. Další negativní vlivy dronů jsou možné nehody se smrtícím potenciálem (pokud na člověka spadne několikakilogramový dron, následky mohou být fatální), střetnutí s jiným dronem, letadlem či ptákem a plašení zvěře, kdy ptáci na drony útočí. Technologii dronů lze považovat za ukázkový případ předběhnutí legislativy technikou a v současnosti státy zavádí nové zákony a regulace pro provoz těchto strojů.

Vzniká potřeba se proti výše uvedeným bezpečnostním hrozbám a rizikům aktivně bránit a předcházet jim. Protože je technologie dronů relativně nová a ještě před několika lety nebyly drony tak rozšířeny jako je tomu dnes, je i obrana proti dronům ve svých počátcích. K obraně proti dronům je nutná v první řadě jejich detekce a lokalizace, lokalizace jejich pilotů, dále pak rušení řídicího signálu, oslepení pomocí kouře, sestřelení, zachycení do sítě, spuštění poplachu, evidence událostí a jejich hlášení. K těmto se účelům používají radary, detektory řídicích rádiových signálů (Wi-Fi), mikrofony, kamery a infrakamery. Za nejlépe rozvinutou technologii lze označit detekci rádiových signálů, kde již existují komerční

řešení. Nevýhodou radaru je obtížná detekce malých dronů, kvůli malé odrazové ploše. Detekci mikrofonom negativně ovlivňuje malý dosah a šum. Infrakamery mají v současnosti malé rozlišení a drony jsou obtížně rozeznatelné díky malému kontrastu oproti okolnímu vzduchu. Na druhou stranu představují řešení pro detekci v noci. Zbývající alternativou je nahrávání obrazu ve viditelném spektru a detekce dronů pomocí algoritmů počítačového vidění. V průběhu vyhledávání zdrojů jsem narazil na jedno komplexní komerční řešení uvedeného problému. Zahrnuje detekci dronů pomocí rádiových signálů, zvuku, infrazáření a také obrazu zpracovaného klasifikační konvoluční neuronovou sítí. Tyto vstupy pak umožňuje propojit s rozhodovací logikou, aktivním hlášením událostí (např. SMS, email), informačním systémem a se stávajícím bezpečnostním systémem.

Cílem této práce je analýza současných metod pro detekci pohybujících se objektů ve videu nahraném pohybující se kamerou. Tento úkol není jednoduchý, protože je nutné vyřešit problémy týkající se dvou rozdílných pohybů ve videu. Dalšími cíli jsou návrh, implementace, testování a vyhodnocení systému pro detekci dronu či hejna dronů. Implementovaná metoda bude pracovat v reálném čase, zdrojem bude video nebo výstup z kamery. Poté bude metoda otestována jak na offline videu tak na video výstupu z kamery přímo v terénu se skutečnými drony. Závěrem bude vyhodnocena úspěšnost detekce.

Tato práce je součástí projektu řešeného ve spolupráci s Univerzitou obrany v Brně. Jejím přínosem bude možnost začlenění do projektu a pro seznámení čtenáře se současnými metodami pro detekci pohybujících se objektů.

Práce je dále členěna do pěti kapitol: druhá kapitola obsahuje úvod do problematiky detekce a sledování objektů, problémy s tím spojené a metriky, které se používají pro vyhodnocení detekčních algoritmů. Třetí kapitola popisuje současné metody pro detekci pohybujících se objektů jako optický tok, konvoluční neuronové sítě a kosegmentace. Návrh systému je popsán ve čtvrté kapitole. Poslední dvě kapitoly popisují implementaci navrženého systému a jeho vyhodnocení. V rámci závěru jsou diskutovány možnosti budoucího vývoje.

Kapitola 2

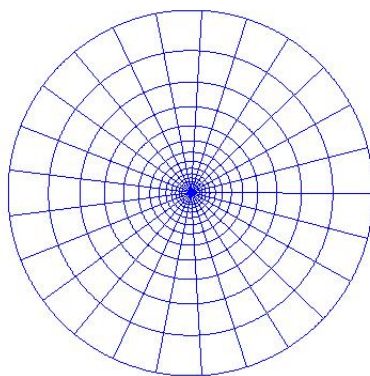
Přehled současného stavu

V této kapitole jsou popsány základní pojmy zpracování obrazu, které je zapotřebí znát pro porozumění následujícím kapitolám a principům detekce objektů ve videu. V kapitole jsou vysvětleny pojmy reprezentace digitálního obrazu, předzpracování obrazu, detekce objektů a sledování objektů, dále pak výzvy a problémy, týkající se detekce pohybujících se objektů a metriky, které se běžně používají pro vyhodnocení kvality detekčních algoritmů.

2.1 Reprezentace obrazu

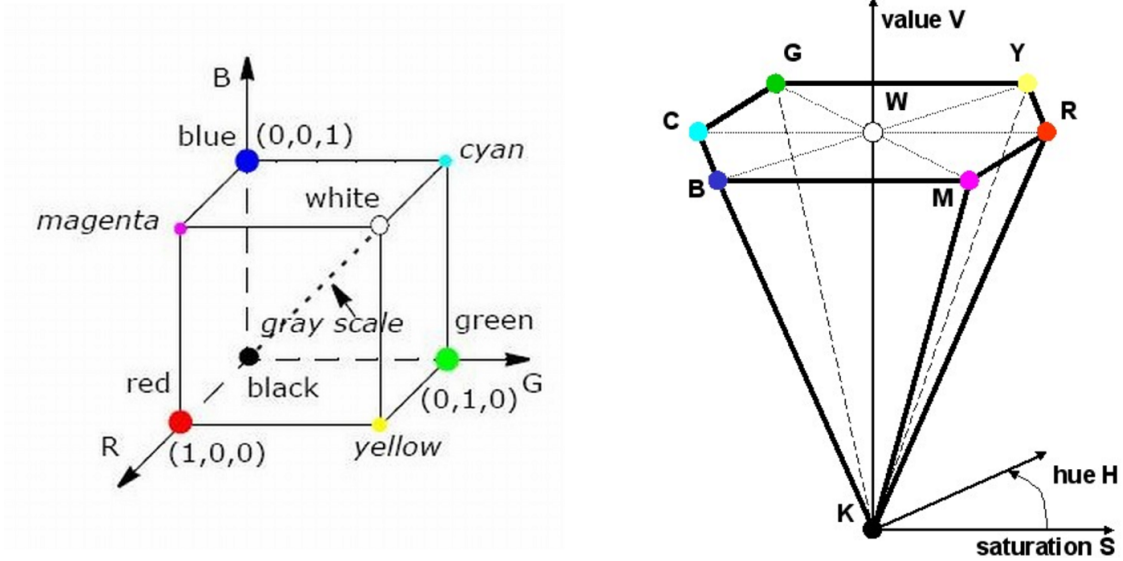
Obraz je grafická informace definovaná dvourozměrnou funkcí $f(x, y)$, kde x a y jsou prostorové souřadnice a f je hodnotou v tomto bodě. Hodnota f může být buď skalár, kdy popisuje jas neboli intenzitu bodu, anebo vektor, kdy jednotlivé složky vektoru jsou definovány barevným modelem. U digitálního obrazu jsou všechny veličiny disktrétní. Bod obrazu se nazývá pixel.

Nejčastěji používaná soustava souřadnic je kartézská soustava souřadnic, kde osy jsou vzájemně kolmé přímky, které se protínají v počátku. Pro algoritmy počítačového vidění lze využít i jiné soustavy souřadnic, např. logaritmickou polární soustavu, kde je bod určen dvěma čísly, první je logaritmus vzdálenosti od počátku a druhé je úhel. Tuto soustavu použil Zhang et al. [39] pro urychlení metod optického toku.



Obrázek 2.1: Příklad logaritmické polární soustavy.

Barvy pixelu jsou popsány barevnými modely. Nejpoužívanějším barevným modelem je model RGB (Red, Green, Blue), kde je výsledná barva složena ze tří základních barev a to červené, zelené a modré, jedná se tedy o aditivní model. Každá barevná složka je popsána určitým počtem bitů, tento vztah se nazývá barevná hloubka. Nejčastěji se používá 24 bitů na pixel, tato barevná hloubka se označuje termínem True Color. Pro některé algoritmy může být vhodnější barevný model HSV (Hue, Saturation, Value), kde první složka je odstín (Hue), druhá sytost barvy (Saturation) a třetí hodnota jasu (Value). Díky oddělení jasu od barvy, je možné řešit problémy spojené s náhlou změnou osvětlení [19] nebo s odstraněním stínu [30].



Obrázek 2.2: Barevné modely RGB (vlevo) a HSV [25].

Mnoho algoritmů nepotřebuje zpracovávat barevnou informaci, proto pracují se šedotónovým obrazem, kde jas je popsán odstíny šedi. Příklady algoritmů vyžadujících šedotónový obraz jsou detektory hran, které pracují s náhlou změnou jasu, nebo optický tok, kde se pohyb v obraze určuje podle jasu jednotlivých pixelů. Výhodou jsou také menší výpočetní nároky než u barevného obrazu. Šedotónový obraz můžeme získat z šedotónové (jednobarvné – mono-color) kamery nebo převodem z barevného modelu. Pro převod z modelu RGB se používá vzorec

$$I = 0,299R + 0,587G + 0,114B \quad (2.1)$$

kde I je výsledný odstín šedi (jas) a R, G, B jsou barevné složky modelu RGB [25]. Jednotlivé barvy jsou v součtu váhovány, protože lidské oko je na barvy různě citlivé.

2.2 Předzpracování obrazu

Před samotnou aplikací algoritmů počítačového vidění je často nutné zdrojový obraz upravit tak, aby co nejvíce vyhovoval následujícím algoritmům v posloupnosti zpracování. Tyto úpravy se liší podle použitých algoritmů.

Standardní úpravou je změna velikosti obrazu, tedy zvětšení/zmenšení počtu řádků/sloupců v obraze. Zmenšením vstupního obrazu se obecně dosahuje zrychlení algoritmů, protože klesají časová a paměťová náročnost algoritmů, ovšem na úkor přesnosti. Algoritmy počítačového vidění jsou obvykle náročnější na výpočet, standardní časové složitosti jsou lineární $O(n \log n)$, kvadratická $O(n^2)$ a kubická $O(n^3)$.

Další velmi často používanou úpravou je v předchozí podkapitole zmíněný převod barevného na šedotónový obraz 2.1. Podobnou úpravou je převod šedotónového obrazu na binární (černobílý). Jednou z metod pro tuto úpravu je prahování, kde je hodnota jasu pixelu porovnávána s určeným prahem (threshold). Pokud je hodnota jasu menší než práh, pixel bude černý, jinak bílý. Tato jednoduchá metoda je často součástí mnoha algoritmů, např. u odčítání pozadí a dá se použít pro odstranění odlehlých hodnot (outliers). Obecný vzorec pro prahování je:

$$f(x) = \begin{cases} A & \text{pro } x < T \\ B & \text{pro } x \geq T \end{cases} \quad (2.2)$$

kde x je vstupní hodnota, $f(x)$ je výsledná hodnota, T je práh, A je nová hodnota pro x pod prahem a B je nová hodnota pro x nad prahem.

Pro vyhlazení obrazu se velmi často využívá Gaussova vyhlazování. Tato metoda redukuje šum a rozmazává hrany. Jedná se o konvoluční filtr, kde konvoluční jádro jsou váhy dané Gaussovou funkcí. Vzorec pro Gaussovo vyhlazování je:

$$h(x, y) = f(x, y) * G(x, y) \quad (2.3)$$

kde $f(x, y)$ je vstupní pixel, $h(x, y)$ je výsledný pixel a $G(x, y)$ je Gaussova funkce:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2.4)$$

2.3 Detekce objektů

V následujících řádcích bude definován termín detekce objektů ve videu. Video je posloupnost obrazů, kde každý z těchto obrazů se nazývá rámeček. Obsah dvou po sobě jdoucích rámečků spolu obvykle souvisí. Detekce objektu(ů) je zjištění přítomnosti a polohy objektu ve videu [14]. Tuto definici lze rozšířit na nalezení nejmenšího obdélníku ohraničující objekt [2].

Metody pro detekci pohybujících se objektů lze rozdělit na tři hlavní kategorie. Metody první kategorie spoléhají na vzhled objektů v jednotlivých snímcích, druhé kategorie spoléhají na informaci o pohybu mezi rámci a metody třetí kategorie kombinují obě předchozí metody [28].

2.3.1 Metody založené na vzhledu

Tyto metody pracují v prostorové doméně (spatial domain). Zahrnují informace obsažené pouze v jednom obraze, lze je tedy využít i pro detekci objektů z fotografie. Každý objekt má své charakteristické rysy (features), které ho popisují. Těmito rysy mohou být rohy, hrany, barvy, regiony, tvary nebo textury. Deskriptor popisuje jeden nebo více rysů. Příklady deskriptorů jsou průměr (např. průměrná barva) nebo histogram (histogram orientovaných



Obrázek 2.3: Příklad detekce objektu, objekt je ohraničený obdélníkem (bounding boxem) [15].

gra dientů). Jako příklady algoritmů extrahující deskriptory lze uvést Speeded Up Robust Features (SURF) [4] a Histogram of Oriented Gradients (HOG) [9].

Následně je na deskriptorech natrénován klasifikátor, který zařadí objekt do příslušné třídy. Často používané klasifikační algoritmy jsou AdaBoost [11] a Support vector machines (SVM) [8]. Poté se obvykle použije metoda Sliding window, kde se vytvoří okno nad obrazem, které se postupně posouvá. Pokud klasifikátor vyhodnotí, že se v okně nachází rozpoznávaný objekt, vrátí se okno jako bounding box (ohraničující obdélník) [2].

Dalším přístupem založeným na strojovém učení jsou Konvoluční neuronové sítě (CNN – Convolutional Neural Networks). Od předchozích dvou algoritmů se liší tím, že deskriptory jsou získány přímo pomocí CNN [2].

Nevýhodou algoritmů založených na strojovém učení je nutnost mít rozsáhlou trénovací množinu dat. Pro běžné aplikace jako detekce obličejů, chodců nebo automobilů existují trénovací množiny volně dostupné na Internetu, ovšem pro detekci dronů téměř neexistují, protože se jedná o relativně nový problém. Pánové Aker a Kalkan [2] tento problém vyřešili vytvořením vlastní umělé trénovací množiny. Další nevýhodou je, že tyto algoritmy pracují nejlépe, když jsou hledané objekty dostatečně velké a jasně viditelné. Je velmi obtížné detekovat malý objekt, což je žádoucí u detekce dronů [28].

Alternativou k metodám strojového učení je kosegmentace [27] [23]. Tato metoda segmentuje současně společnou část více obrazů [3]. Kosegmentace je speciální případ segmentace, kdy na základě vyhledávacího obrazu (query image) je v jiném obrazu nebo rámci videa nalezena společná část obrazů. Obrovskou výhodou této metody je to, že kromě vyhledávacího obrazu nepotřebuje žádnou předchozí informaci ani lidský dohled [3]. V porovnání s metodami strojového učení odpadá trénování klasifikátorů na tisících příkladů.

2.3.2 Metody založené na pohybu

Metody založené na pohybu pracují v časové doméně (temporal domain). Pohybující se objekt lze popsat jako množinu pixelů v rámci videa mající souvislý pohyb v čase a sémantickou podobnost v prostoru obrazu [38]. Pokud detekce zahrnuje pouze informaci o pohybu, detekují se všechny pohybující se objekty ve videu. To je nežádoucí, protože spolu s cílovým objektem (např. auto) se detekují i ostatní objekty jako kymácející se stromy, plovoucí oblaka apod.

Metoda odčítání pozadí pracuje tak, že se od jednotlivých pixelů aktuálního rámce odečítají pixely pozadí. Výsledkem metody je maska změn v obraze, ze které je možné vysegmentovat objekt. Pro zlepšení výsledku se používá prahování. Tato metoda je velmi rychlá, je však poměrně nepřesná, protože je založená na hypotéze statického pozadí. Proto není vhodná pro video zachycené pohybující se kamerou.

Jinou široce používanou metodou je optický tok. Optický tok je rozdělení zjevných rychlostí pohybu jasových vzorů v obraze [16]. Lze si jej představit jako 2D pole vektorů, kde vektory vyjadřují směr a velikost pohybu jednotlivých pixelů mezi dvěma obrazy. Tato metoda je vhodná pro video zachycené pohybující se kamerou, je ovšem poměrně výpočetně náročná. Proto se optický tok běžně nepočítá pro všechny body v obraze, ale pouze pro podmnožinu charakteristických bodů jako jsou rohy [26].

2.3.3 Hybridní metody

Hybridní metody kombinují informace o vzhledu a pohybu objektu, pracují v prostorovo-časové doméně (spatio-temporal domain). Teoreticky jsou hybridní metody nejlepším přístupem pro detekci objektů, protože pracují s více informacemi než metody předchozích dvou kategorií. Práci implementujících tyto metody je výrazně méně než těch pracujících v prostorové nebo časové doméně. Příkladem je [28], kde je použit koncept st-cubes (spatio-temporal cubes – prostorovo-časové kostky), které jsou složeny z několika rámců videa. Dále je kompenzován pohyb objektů mezi snímky a použit klasifikátor AdaBoost. Jako další příklad je možné uvést [35], kde autoři kombinují optický tok, SVM algoritmus a konvoluční neuronové síť.

Některé z výše uvedených algoritmů budou podrobně vysvětleny v následující kapitole. Většinou je detektor objektů tvořen více než jedním algoritmem a algoritmy jsou kombinovány a různě modifikovány tak, aby co nejlépe plnily účel výsledné aplikace (např. detekce obličeje). Obecně se dá říci, že čím více informací je z videa extrahováno, tím přesnější je detekce objektu. S tím je ovšem spojena vyšší výpočetní náročnost. Výpočetní výkon počítačů se ovšem stále zvyšuje, algoritmy je možné akcelarovat výpočtem na grafické kartě (GPU) nebo pomocí obvodu FPGA (Field Programmable Gate Array). Navzdory desetiletým výzkumům neexistuje dokonalý algoritmus pro detekci objektů, a proto je detekce objektů stále otevřená oblast aktivního výzkumu.

2.4 Sledování objektů

Termín sledování objektů (object tracking) často splývá s termínem detekce objektů, ale tyto termíny se vzájemně liší, ačkoli mají mnoho společného. Sledování objektů je lokalizace pohybujícího se objektu v po sobě jdoucích rámcích videa [38]. Metody sledování objektů lze rozdělit do dvou hlavních kategorií: klasické sledovací metody a sledování detekcí.

Obecný koncept klasických sledovacích metod se skládá ze čtyř kroků. Prvním je výběr sledovaného objektu a to buď uživatelem, nebo automaticky detekčním algoritmem. Druhým krokem je extrakce rysů objektu. Třetím krokem je nalezení nejlepšího shody v následujícím rámci pomocí extrahovaných rysů. Posledním krokem je aktualizace rysů podle nalezené shody. Dále se opakují kroky 3 a 4 [38].

Detekce objektů a extrakce rysů byla popsána v předchozí podkapitole. K nalezení nejlepší shody se používají algoritmy založené na pravděpodobnosti. Jako příklady lze uvést metody založené na kernelu (Mean-Shift [7]), kde se počítá pravděpodobnost výskytu barvy

pixelu objektu v předchozím rámci. Poté se prohledává okolí předchozí pozice objektu, výsledná pozice je dána největší pravděpodobností. Jinou metodou je částicový filtr (particle filter), částice (vzorky) jsou obdélníky reprezentující hypotetické umístění sledovaného objektu, které jsou rozmístovány náhodně v okolí předchozí pozice objektu. V obdélnících je poté vypočtena podobnost pomocí histogramu barvy. Částice s největší podobností určují polohu objektu. Tato metoda byla podrobně popsána a použita v [24]. Pro sledování objektů se také používají metody založené na Kalmanově filtru nebo Konvolučních neuronových sítích.

Sledování detekcí (tracking-by-detection) spočívá v detekování objektu v každém rámci videa. Tento přístup byl zvolen v [3].

2.5 Problémy spojené s detekcí pohybujících se objektů

Detekce pohybujících se objektů ve video sekvencích zachycených pohybující se kamerou je spojená s mnoha problémy. Pánové Yazdi a Bouwmans [38] shrnuli ve své práci hlavní problémy. V této podkapitole budou tyto problémy stručně popsány a uvedeny do kontextu s detekcí dronů.

- **Variance osvětlení** – osvětlení scény a sledovaného objektu se může měnit v čase. Důvody jsou pohyb zdrojů světla, odraz světla, pokles intenzity zdroje světla, ve venkovních scénách počasí, zakrytí zdroje světla atd. Důsledkem může být selhání metod založených na vzhledu.

Detekce dronů bude probíhat výhradně ve venkovních scénách. Velkou roli zde hraje počasí, hlavně oblačnost má největší vliv na světelné podmínky. Přechod mraků způsobuje prudkou změnu osvětlení. Dále může dojít k odrazu světla od střech, vodních ploch, oken nebo skel automobilů.



Obrázek 2.4: Příklad variance osvětlení. Na snímcích jsou patrné změny vlivem denní doby a počasí².

- **Změna vzhledu pohybujícího se objektu** – video zachycuje pouze 2D projekci 3D scény, takže jakákoli rotace podél třetí osy může změnit vzhled objektu. Jiné změny vzhledu objektů mohou být: změna výrazu obličeje, roztáhnutí křídel, nafouknutí balónu, změna oblečení atd.

Tento problém je u detekce dronů minimální, naprostá většina dronů svůj vzhled za letu nijak nemění.

²Zdroj: <https://www.youtube.com/watch?v=VSaFnWeUHPo>, nezměněno



Obrázek 2.5: Příklad změny pohledu objektu - skládací dron³.

- **Prudký pohyb** – prudké změny rychlosti a směru pohybu objektu nebo kamery můžou způsobit ztrátu objektu sledovací metodou (trackerem) nebo velkou chybu v odhadu polohy objektu.

Drony jsou schopny prudkého pohybu, většinou ale ne tak zásadního, aby způsobil ztracení trackerem. To je ale možné při poryvu silného větru.



Obrázek 2.6: Příklad prudkého pohybu - srážka dvou aut⁴.

- **Okluze** – sledovaný objekt může být částečně nebo zcela zakryt jiným objektem, např. stromy, vozidly, budovami nebo chodci.

Drony můžou být zakryty zejména vegetací, terénem a v obydleném území navíc budovami a případně dopravními prostředky.

- **Komplexní pozadí** – ve scénách s komplexním pozadím může mít i lidské vidění problém s detekcí objektu. Platí to hlavně pro venkovní scény. V pozadí se také může vyskytovat pohyb, jako kymácející se stromy, plovoucí oblaka, vodní vlny atd. Díky tomu některé algoritmy nemusí objekt detekovat.

Pro detekci dronů může být tento problém kritický. Zvláště může být obtížné detekovat dron ve velké vzdálenosti, kdy je ve videu pouze na několika desítkách pixelů. Dron může splývat s pozadím - např. šedý dron na šedém mraku nebo dron na pozadí s hustou vegetací.

³Zdroj: http://rpg.ifi.uzh.ch/docs/RAL18_Falanga.pdf, nezměněno

⁴Zdroj: <https://www.youtube.com/watch?v=X7dtS1BIePQ>, nezměněno



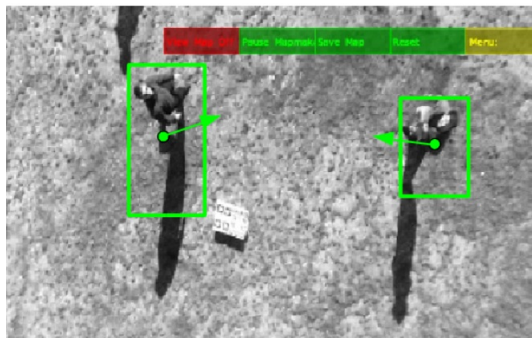
Obrázek 2.7: Příklad okluze objektu. Auto je částečně zakryto pilířem mostu.



Obrázek 2.8: Příklad komplexního pozadí [28].

- **Stín** – mnoho algoritmů nedokáže odlišit objekt od jeho stínu a detekuje jej jako součást objektu, bounding box tak obsahuje jak objekt, tak jeho stín. Tento problém se vyskytuje zejména u video sekvencí zachycených z výšky.

Pro detekci dronů stín nepředstavuje žádný problém, protože stín dronu je velmi malý a většinou mimo záběr kamery.



Obrázek 2.9: Příklad přítomnosti stínu v obraze [26]. Bounding box osoby vlevo zahrnuje velkou část stínu.

- **Problémy spojené s kamerou** – kamera je důležitou součástí systému pro detekci objektů. Kamery mají různé senzory, čočky, rozlišení, rychlost snímání a vnitřní obvody pro zpracování obrazu. Nekvalitní video sekvence může znemožnit detekci objektů. Dalšími problémy jsou rozmazání vznikající vibracemi kamery a blokové ar-

tefakty vznikající při kompresi videa. Kvalitu video sekvence také výrazně zhoršuje přítomnost šumu.



Obrázek 2.10: Příklad nekvalitního obrazu⁵.

- **Pohyb kamery** – Detekce objektů ve videu zachyceném pohybující se kamerou je složitější než ve videu zachyceném stabilní kamerou. Je nutné rozpoznat pohyb kamery od pohybu objektů a kompenzovat pohyb kamery. Tento úkol ovšem není jednoduchý, kamera se může pohybovat ve směru všech tří os prostoru. Pohyblivé bezpečnostní kamery umožňují jednoduchý pohyb PTZ (Pann – Tilt – Zoom). Složitější je řešení u kamer připevněných na létající prostředky (dron, letadlo), kde se kamera může pohybovat libovolně v 3D prostoru.

V této práci bude použita pozemní kamera s možností PTZ pohybů.



Obrázek 2.11: Příklad obrazu zachyceného pohybující se kamerou, zde připevněnou ke dronu [15].

- **Deformace nepevných objektů** – nepevný pohybující se objekt je takový objekt, jehož části mohou mít různý pohyb. Příkladem je chodec, který při chůzi pohybuje končetinami, nebo pták mávající křídly. Mnoho algoritmů pak detekuje jednotlivé části objektu jako rozdílné objekty.

Mezi nepevné objekty se řadí i drony, kde vrtule dronu mají rozdílný pohyb od dronu jako celku.

⁵Zdroj: <https://cz.webcams.travel/webcam/fullscreen/1206291667>, nezměněno



Obrázek 2.12: Příklad deformace nepevných objektů.

2.6 Metriky pro vyhodnocení detektorů

Tato podkapitola byla částečně převzata z práce pánů Yazdih a Bouwmanse [38]. Pro vyhodnocování a porovnávání algoritmů pro detekci pohybujících se objektů je zapotřebí stanovit kvantitativní metriky, které vyčíslují kvalitu algoritmu. Pro použití metrik je nutné mít tzv. *ground-truth* obrazy, zjednodušeně binární masky objektů v obraze. Objekt je v *ground-truth* obraze vyznačen bílými pixely, zatímco pozadí černými, viz obr 2.13. Problémem je získání takovýchto masek, obvykle se musí vytvářet ručně, což je pro velké množiny dat zdlouhavé. *Ground-truth* obraz umožňuje použití obvyklých metrik pro binární klasifikaci dle následujících parametrů:

- **True Positive (TP)** – počet pixelů detekovaného objektu odpovídajících detekovaným pixelům v *ground-truth*.
- **False Positive (FP)** – počet pixelů detekovaného objektu odpovídajících nedetekovaným pixelům v *ground-truth*.
- **True Negative (TN)** – počet pixelů nedetekovaného objektu (pozadí) odpovídajících nedetekovaným pixelům v *ground-truth*.
- **False Negative (FN)** – počet pixelů nedetekovaného objektu (pozadí) odpovídajících detekovaným pixelům v *ground-truth*.

Z výše uvedených metrik se skládají nejpoužívanější metriky:

- **Precision** – měří procento všech detekovaných pixelů náležícím objektu.

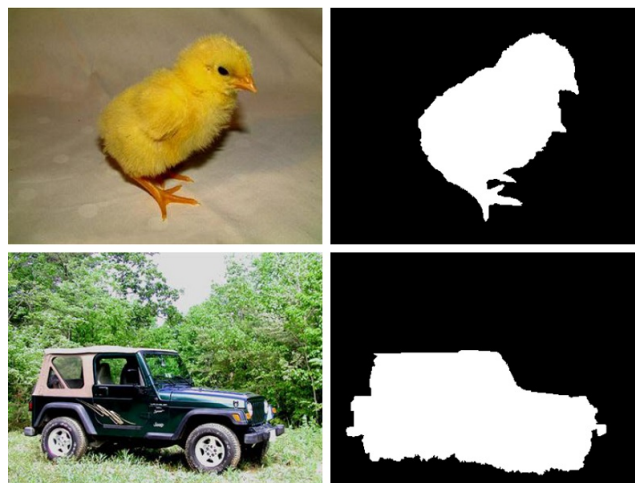
$$Precision = \frac{TP}{TP + FP} \quad (2.5)$$

- **Recall** – měří procento všech pixelů náležícím objektu, které jsou správně detekovány.

$$Recall = \frac{TP}{TP + FN} \quad (2.6)$$

Jinou používanou metrikou je **Accuracy**, která měří procento všech pixelů, které jsou správně detekovány a odmítnuty:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (2.7)$$



Obrázek 2.13: Příklady ground-truth obrazů: vlevo originální obrazy, vpravo jejich ground-truth obrazy [6].

Pro metody sledování objektů, které reprezentují objekt bounding boxem se používá **Intersection over union (IOU)**, což je poměr přesahující oblasti předpokládaného bounding boxu a ground-truth bounding boxu k sjednocení obou oblastí:

$$\text{Intersection over union} = \frac{A \cap GT}{A \cup GT} \quad (2.8)$$

kde A je oblast předpokládaného bounding boxu a GT oblast ground-truth bounding boxu. Metriku lze také použít pro vyhodnocení segmentace/kosegmentace. V některých pracích je tato metrika nazvána overlap ratio.

Často je autory používána reprezentace výsledků pomocí křivky. Nejpoužívanějším křivkou je **Precision-Recall (PR)**. Precision-Recall křivka je vytvořena vykreslením Precision proti Recall, příklady na obr 2.14. Čím více je křivka blíže k pravému hornímu rohu, tím lepší je výkon metody [2].

Jednotliví autoři používají různé vlastní metriky, často vytvořené úpravou výše popsaných metrik. Například Rozantsev, Lepetit a Fua [28] spojili Precision a Recall do *Average Precision (AveP)* podle vzorce:

$$\text{Average Precision} = \int_0^1 p(r) dr \quad (2.9)$$

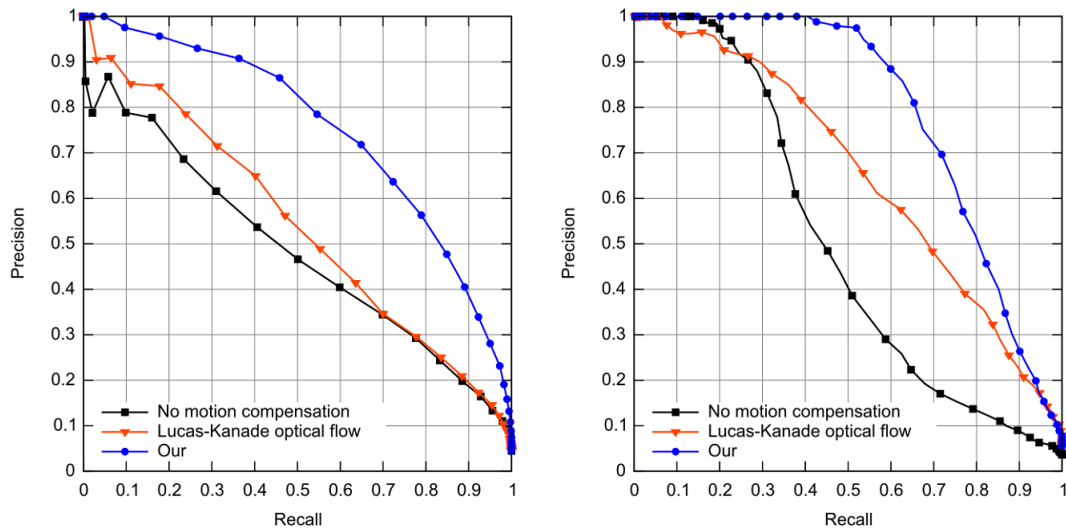
kde p je Precision a r je Recall.

Bandyopadhyay [3] použil metriku *Error-rate*, která je vypočítána vzorcem:

$$\text{Error - rate} = 1 - \frac{A \cap GT}{A \cup GT} \quad (2.10)$$

jedná se tedy o inverzní hodnotu Intersection over union.

V některých pracích autoři upravují definice základních metrik, např. Aker a Kalkan [2] považují předpokládaný bounding box jako True Positive, když je Intersection over union



Obrázek 2.14: Příklady Precision-Recall (PR) křivek [28].

větší než 50 %. Při vyhodnocování a porovnávání s jinými pracemi autoři často uvádějí použité množiny dat (datasets), počet obrazů, videí a rámců videí, jejich rozlišení a případně i velikost objektů v obrazech. Výsledky metod pro detekci objektů závisí na vstupních datech, zvolených kritériích pro vyhodnocení a při vyhodnocení potřebného času na použitém hardwaru. Metoda *A* dávající lepší výsledky než metoda *B* na určité množině dat může na jiné množině dat dávat horší výsledky. Důležité je při porovnávání používat stejnou datovou množinu a stejný formát dat pro všechny porovnávané metody. Otázkou je, jestli je možné mít vždy lepší výsledky, pokud se zvolí ty „správné“ množiny dat a porovnává se proti „správným“ pracím.

Autoři ve svých pracích většinou nevyhodnocují časovou náročnost metod, ale pokud tak učiní, uvádí některé z těchto údajů: časovou složitost, průměrnou dobu detekce pro jeden obraz (rámec), počet snímků za sekundu (FPS) a údaje o použitém hardwaru jako model a frekvence procesoru a velikost operační paměti.

2.7 Klasifikační algoritmy

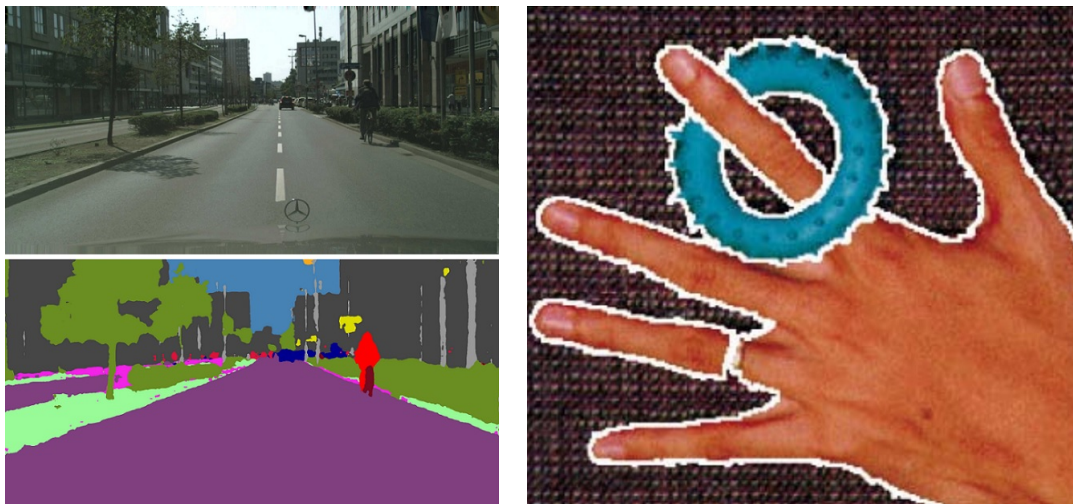
2.8 Konvoluční neuronové sítě

2.9 Segmentace

Segmentace je proces rozdělení obrazu do několika segmentů (superpixelů). Cílem je zjednodušit nebo změnit reprezentaci obrazu pro snazší analýzu [29]. Pixely patřící jednotlivým segmentům spolu sdílí určité vlastnosti jako barva, jas nebo textura. Segmentaci lze také popsat jako proces přiřazení labelu (štítku) každému pixelu v obraze. Aplikací segmentace je velmi mnoho: detekce objektů, vyhledávání obrazů, detekce a rozpoznání tváře, rozpoznávání otisků prstů, dohledové systémy, strojové vidění, atd.

Metody segmentace se dělí na dvě hlavní skupiny – segmentace s dohledem (supervised) a segmentace bez dohledu (unsupervised). Metody segmentace s dohledem mohou na základě předchozí informace dosáhnout sémantické segmentace – tedy segmentace konkrétního objektu. Předchozí informace (object prior) je získána metodami jako např. detekce

významného objektu (saliency object detection), metody strojového učení (AdaBoost, SVM, CNN) a metody s lidskou interakcí. Naopak metody segmentace bez dohledu segmentují obraz na mnoho jednotných a homogenních oblastí na základě společných vlastností jako je barva nebo textura [23]. Rozdíl mezi oběma skupinami lze vidět na obr 2.15.



Obrázek 2.15: Příklad segmentace obrazů. Vlevo segmentace s dohledem, vpravo segmentace bez dohledu [33].

Mezi segmentační metody patří metody shlukování (Mean-Shift), metody založené na růstu regionů, metody založené na Konvolučních neuronových sítích, metody aktivních obrysů (Active Contours) založené na hledání hran (známé jako Snakes) nebo na Level set funkci a metody založené na Markovských náhodných polích. Metoda aktivních obrysů s Level set funkcí a metody založené na Markovských náhodných polích jsou popsány v rámci podkapitoly Kosegmentace 2.17.

Podkategorií segmentace je segmentace video objektů, která je definována jako segmentace objektu napříč rámci video sekvence. Na rozdíl od segmentace jednoho obrazu zde lze využít pohybu objektu mezi rámci pro kvalitnější výsledky. Takové metody jsou označovány jako hybridní metody, příkladem je [35], kde je využit optický tok.

2.9.1 Sémantická segmentace založená na Konvolučních neuronových sítích

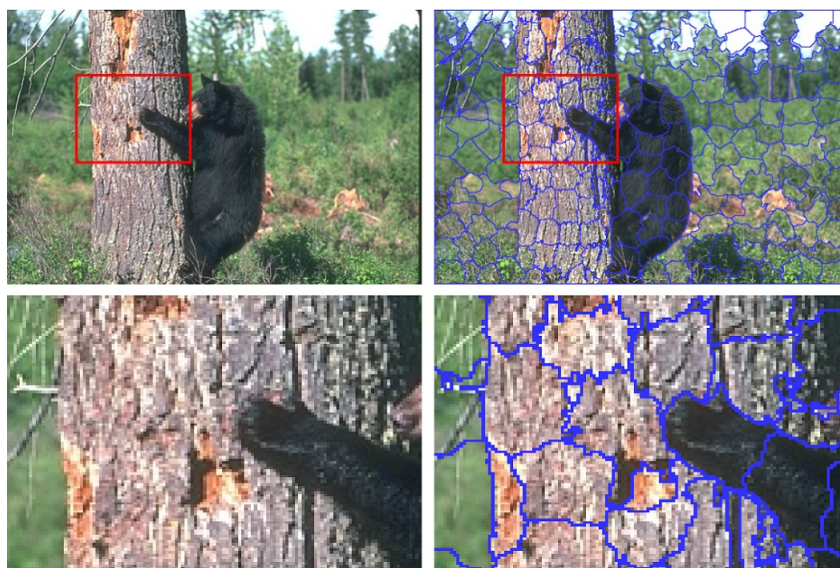
V posledních letech se pro sémantickou segmentaci začali široce používat Konvoluční neuronové sítě (CNN), kvůli jejich schopnosti extrahovat kompaktnější a smysluplnější rysy z obrazu než klasické algoritmy (např. HOG). Ukazuje se, že CNN dosahují jasně lepších výsledků než starší přístupy [12].

Jedním z přístupů jak dosáhnout lepších výsledků je použití pravděpodobnostních modelů jako Markovská náhodná pole a Conditional Random Fields (CRF) [12]. CRF jsou variantou Markovských náhodných polí. Příkladem kdy je k poslední vrstvě připojeno CRF je DeepLab [5]. Vysoká přesnost se obvykle pojí s delším časem nutným pro segmentaci. Zhao *et al.* [40] proto navrhli Image Cascade network (ICNet), která dosahuje 30 FPS na obrazech s rozlišením 1024×2048 a zároveň vysoké přesnosti. Principem sítě je použití kaskády vstupů, kde je vstupem trojice obrazů v nízkém, středním a vysokém rozlišení.

Využívá se tak efektivně sémantické informace v nízkém rozlišení spolu s detaily z vysokého rozlišení.

2.9.2 Texture-Aware Superpixel Segmentation

Jedna z posledních prací zaměřující se na segmentaci bez dohledu je Texture-Aware Superpixel Segmentation [13]. Giraud *et al.* vytvořili metodu, která segmentuje obraz nejen na základě barvy, ale bere v potaz i texturu. Metoda je vylepšením metody SLIC [1] a je založená na k-means shlukování. K výrazu pro prostorovou vzdálenost a výrazu pro barvu přidává výraz pro průměrnou vzdálenost pixelů s podobnou texturou okolí pixelu. Ve výsledku je jeden shluk segmentem (superpixelem) v obraze. Výhodami této metody jsou univerzálnost, nízká výpočetní náročnost a malý počet parametrů.



Obrázek 2.16: Výsledky metody Texture-Aware Superpixel Segmentation na vysoce texturovaných obrazech [13]. Vlevo vstupní obraz, vpravo výsledná segmentace.

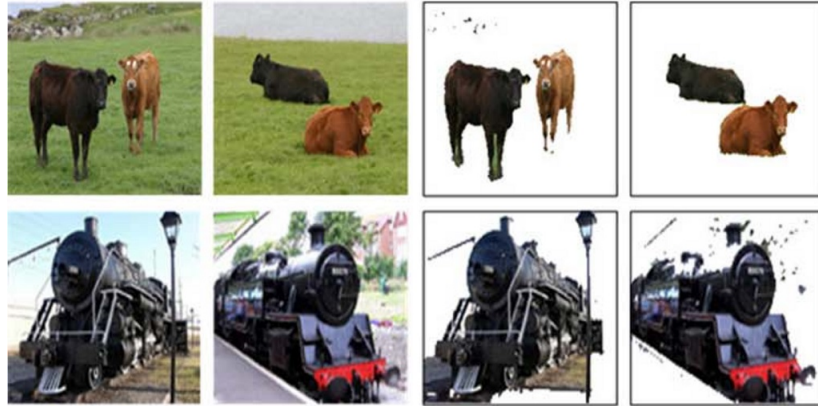
2.10 Kosegmentace

Kosegmentace (cosegmentation) je speciální případ segmentace (2.15), kdy je současně segmentována společná část dvou a více obrazů [3][23][34]. Metoda segmentuje objekty z vícera obrazů extrahováním společných objektů mezi obrazy [23], viz obr 2.17. Kosegmentaci lze použít pro detekci objektů, jako v této práci. Techniky kosegmentace lze rozdělit na techniky založené na Markovských náhodných polích (MRF) a ostatní. Mezi ně patří metody založené na shlukování [18] a také Active contour model. Kosegmentace má mnoho potenciálních aplikací jako klasifikace obrazů, rozpoznávání objektů a vyhledávání obrazů.

Metoda příbuzná kosegmentaci je kosegmentace video objektů (video object cosegmentation), která separuje objekty od pozadí ve videu a to buď interaktivně, nebo automaticky [37]. Tato metoda může fungovat s jedním nebo více videi. Příklady prací jsou [37], kde je použit prostorově-časový auto-kontext model kombinovaný s modelováním vzhledu pro označování superpixelů (segmentů) a [36], kde jsou extrahovány objekty, poté sledovány

napříč videi a kosegmentovány s využitím algoritmů SVM a CNN. Metody popsané v obou pracích jsou hybridními metodami.

Kosegmentaci podobná metoda je kolokalizace, která se zaměřuje na nalezení bounding boxu ohraničujícího společné objekty ve více obrazech [34].



Obrázek 2.17: Příklad kosegmentace. Vlevo původní dvojice obrazů, vpravo výsledky metody [23].

2.10.1 Metody založené na Markovských náhodných polích

Markovská náhodná pole (Markov Random Field – MRF) se používají nejen pro kosegmentaci, ale i pro mnoho dalších úloh v oblastech počítačového vidění a počítačové grafiky jako segmentace, syntéza textur, registrace obrazu a komprese obrazu. Jedná se o pravděpodobnostní model, který je generalizací Markovských procesů [20].

Markovské náhodné pole

Markovské náhodné pole nebo také Markovská síť je definováno jako graf $G = (V, E)$. $V = \{1, 2, \dots, N\}$ je množina uzlů, každý z nich je spojen s náhodnou proměnnou u_j , $j = 1 \dots N$. E značí množinu hran mezi uzly. Okolí uzlu i , značeno \mathcal{N}_i je množina uzlů přilehlých k uzlu i , tedy $j \in \mathcal{N}_i$ pokud a pouze když $(i, j) \in E$. Graf G je označován jako Markovské náhodné pole pokud a pouze když jsou splněny následující podmínky:

$$p(u) > 0, \forall u \in U \quad (2.11)$$

kde $p(u) = p(U = u)$ je vícerozměrná pravděpodobnost, U je skupina náhodných proměnných definovaných v rámci náhodného pole. Platí, že všechny pravděpodobnosti jsou kladné, jedná se tedy o podmínku Pozitivity.

Druhá je podmínka Markovianity:

$$p(u_i | \{u_j\}_{j \in V \setminus i}) = p(u_i | \{u_j\}_{j \in \mathcal{N}_i}) \quad (2.12)$$

kteřá popisuje, že náhodná proměnná u_i závisí jen na sousedních uzlech [20][17].

Labeling problém

Kosegmentace i segmentace je zde definována jako labeling problém, kdy je každému pixelu přiřazen label (štítek) [32]. Jednotlivé pixely, segmenty nebo rohy představují *místa*, množina míst S je:

$$S = \{1, \dots, N\}. \quad (2.13)$$

Každý obraz má množinu labelů L :

$$L = \{1, \dots, M\}. \quad (2.14)$$

Label může popisovat intenzitu pixelu, objekt nebo hranu. V případě kosegmentace i segmentace je množina L binární:

$$L = \{0, 1\}. \quad (2.15)$$

Kde 1 znamená, že pixel náleží objektu a 0 nenáleží objektu.

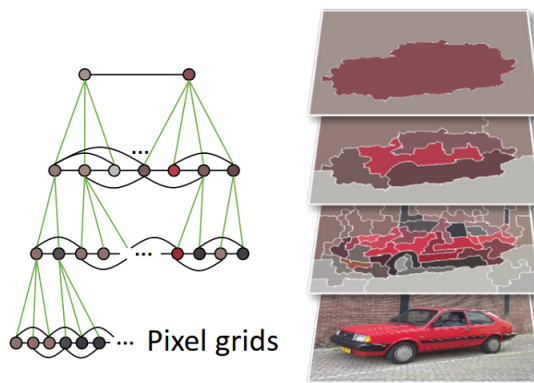
Na problém přiřazení labelu každému místu lze pohlížet jako na hledání zobrazení z S do L :

$$f : S \rightarrow L. \quad (2.16)$$

Model a optimalizace

Pro kosegmentaci je nutné zvolit vhodný model a jeho energetickou funkci. Příklad modelu je na obr 2.18, kde je vidět hierarchický MRF model. Minimalizací energetické funkce hledáme neoptimalnější konfiguraci pole. Mezi optimalizační metody patří graph cut, simulované žíhání nebo belief propagation.

V práci [34] je použit hierarchický MRF model pro kosegmentaci a dense correspondence (hluboká shoda – podobnost obrazů) s využitím algoritmů HOG pro extrakci rysů a graph cut pro optimalizaci energetické funkce. V jedné z posledních prací zabývajících se kosegmentací [22] je použit dynamický MRF model, autoři se zaměřují na detekci a segmentaci objektů v zašuměných videích.

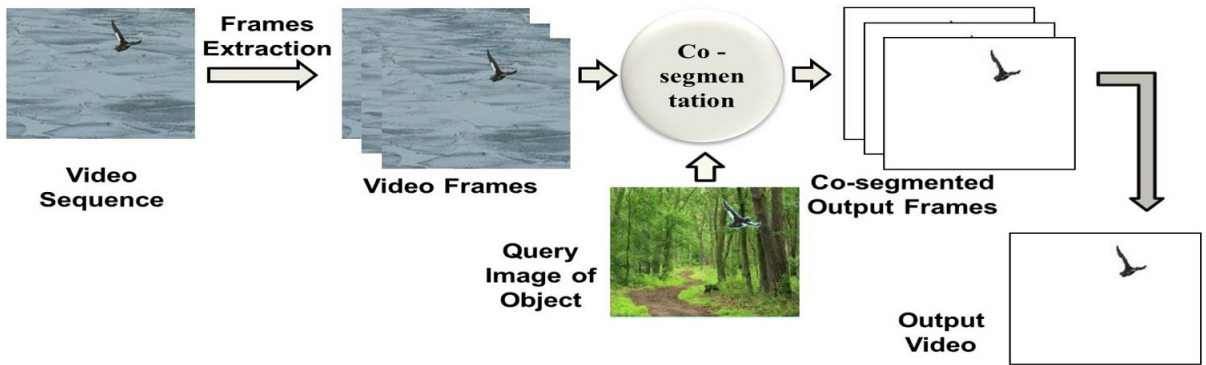


Obrázek 2.18: Ilustrace MRF modelu nad obrazem, nejspodnější vrstva grafu reprezentuje jednotlivé pixely, další vrstvy reprezentují segmenty, které postupně splývají v jeden segment - segmentovaný objekt [34].

2.10.2 Color reward strategy and Active Contour Model

Meng *et al.* [23] navrhl kosegmentační model zahrnující strategii odměňování barev a aktivní obrysový model. Metoda pracuje na principu separace popředí od pozadí aktivním obrysem (křivkou). Je vytvořena energetická funkce se dvěma protichůdnými cíli. Jeden je podobnost popředí mezi obrazy, druhý je konzistence pozadí v každém obraze. Pro reprezentaci regionů je použit barevný histogram a pro měření podobnosti popředí a konzistence pozadí je použita strategie odměn. Energetická funkce je formulována jako level set funkce a iterativně minimalizována.

Aktivní obrysový model je založen na matematickém základě popsáném v mnohokrát citované práci dvojice Chan a Vese [10]. Bandyopadhyay [3] rozšířil uvedenou metodu na detekci objektů ve videu pomocí vyhledávacího obrazu. Princip je následující: vyhledávací obraz obsahuje hledaný objekt (y), video sekvence je rozdělena na jednotlivé rámce a nad každým rámcem je provedena kosegmentace spolu s vyhledávacím obrazem. Výsledkem je detekovaný objekt (y) v každém rámci videa. Výhodou použití kosegmentace pro detekci objektů ve videu je to, že nepotřebuje žádnou předchozí informaci ani lidský dohled.



Obrázek 2.19: Schéma práce navržené Bandyopadhyayem pro detekci objektů ve videu pomocí kosegmentace [3].

Energetická funkce

Energetická funkce pro kosegmentaci založenou na aktivním obrysovém modelu je dána:

$$\begin{aligned}
 E(C_k) = & \mu \cdot Length(C_k) + \nu \cdot Area(C_k^i) \\
 & - \lambda_k^i \int_{C_k^i} f[I_k(x, y), r(C_{1-k}^i)] dx dy \\
 & - \lambda_k^o \int_{C_k^o} f[I_k(x, y), r(C_k^o)] dx dy
 \end{aligned} \tag{2.17}$$

kde I_k , $k \in \{0, 1\}$, je k tý obraz páru obrazů, C_k značí křivku v obraze I_k , regiony uvnitř křivky jsou značeny $r(C_{1-k}^i)$ a regiony vně křivky jsou reprezentovány $r(C_k^o)$. V případě, že pracujeme s *Obrazem 0*, tj. $k = 0$, tak regiony uvnitř křivky jsou $r(C_1^i)$ a regiony vně křivky $r(C_0^o)$. Výraz $f[I_k(x, y), r(C_{1-k}^i)]$ je použit k měření podobnosti mezi pixely v popředí obrazu I_k a regiony uvnitř křivky druhého obrazu $r(C_{1-k}^i)$. Konzistence pozadí je reprezentována výrazem $f[I_k(x, y), r(C_k^o)]$, který měří podobnost mezi pixely pozadí obrazu I_k a vnějších regionů $r(C_k^o)$.

První dva výrazy na pravé straně 2.17 popisují délku křivky a plochu regionů uvnitř křivky. Výrazy $\mu > 0, \nu > 0, \lambda_k^i > 0$ a $\lambda_k^o > 0$ jsou parametry modelu, kde μ je váha pro délku křivky, ν je váha pro plochu regionů uvnitř křivky, váha $\lambda_k^i > 0$ udržuje jednotnost popředí a $\lambda_k^o > 0$ jednotnost pozadí obrazu.

Reprezentace regionů

Regiony jsou reprezentovány normalizovaným barevným histogramem h . Hodnota barvy v RGB obraze je reprezentována 3D vektorem. Histogram je vytvořen s ohledem na pravděpodobnost každé barvy v regionu.

Míra podobnosti

Míra podobnosti f pro hodnotu pixelu p je $f(p, r(C_k)) = h(p)$. Pokud je hodnota pixelu v rozsahu s malou četností hodnot, tj. hodnota má malou pravděpodobnost, míra podobnosti bude malá. Naopak hodnota pixelu v rozsahu s velkou četností vrátí velkou míru podobnosti. Obvyčejně se jednotlivé hodnoty pixelu (barvy) od sebe příliš neliší. Pokud tedy je vzdálenost barev dvou pixelů menší než β , tak budou mít pixely stejnou barvu. Pro hodnotu pixelu p je potom míra podobnosti dána:

$$f(p, r(C_k)) = \sum_{|p' - p| \leq \beta} h(p') \quad (2.18)$$

kde $|p' - p|$ značí největší vzdálenost ze všech vzdáleností mezi hodnotami barevných kanálů RGB.

Formulace level set funkce

Energetická funkce je minimalizována řešením parciální diferenciální rovnice (PDE) level set formulace 2.17. Tedy celková rovnice je změněna na:

$$\begin{aligned} \Phi_k^{n+1}(x, y) = & \Phi_k^n(x, y) + \delta(\Phi_k^n(x, y)) \cdot \left\{ \mu \cdot \operatorname{div} \left(\frac{\nabla(\Phi(x, y))}{|\nabla(\Phi(x, y))|} \right) - \nu \right. \\ & \left. + \lambda_{\Phi_k}^i \cdot f[I_k(x, y), r_{\Phi}(C_{1-k}^i)] - \lambda_{\Phi_k}^o \cdot f[I_k(x, y), r_{\Phi}(C_k^o)] \right\} \end{aligned} \quad (2.19)$$

kde Φ značí level set energetickou funkci křivky. Hodnota level set energetické funkce pro pixel (x, y) v n -té iteraci je reprezentována výrazem $\Phi_k^n(x, y)$. Výraz $\delta(\Phi)$ je Diracova míra definována vzorcem:

$$\delta(\Phi) = \frac{d(H(\Phi))}{d\Phi} \quad (2.20)$$

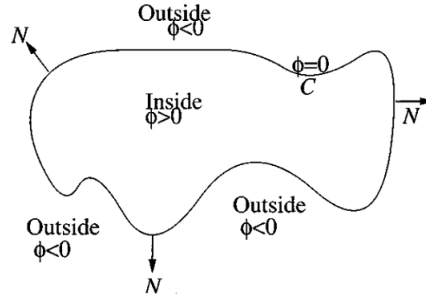
kde $H(\Phi)$ je Heavisideova funkce křivky Φ .

Heavisideova funkce nebo také jednotkový skok je definována jako:

$$H(x) = \begin{cases} 0 & \text{pro } x < 0 \\ 1 & \text{pro } x \geq 0 \end{cases} \quad (2.21)$$

Inicializovaný obrys Φ_k^0 se minimalizací energetické funkce 2.19 začne smršťovat směrem k hranicím objektu (ů). Po určitém počtu iterací se obrys zastaví na hranici, kde je hodnota energetické funkce minimální. Tato hranice separuje společný objekt (y) v obou obrazech a region definovaný křivkou nacházejícím se v rámci videa je považován za hledaný objekt.

Separaci objektu (ů) ilustruje obr 2.20. Z něj je patrné, že křivka C se nachází tam, kde $\Phi = 0$ a dále že pixely s $\Phi > 0$ leží v regionech uvnitř křivky a náleží tedy popředí, zatímco pixely s $\Phi < 0$ leží v regionech vně křivky a náleží pozadí.



Obrázek 2.20: Separace popředí a pozadí pomocí křivky C [10].

Postupný vývoj křivky je zachycen na obr 2.21, kde u dvou párů obrazů je zobrazeno pět stavů vývoje křivky odpovídajících iteracím 1, 101, 201, 301 a 401.

Meng *et al.* [23] použil ve svých experimentech tyto parametry: $\mu = 0,01$, $\lambda_k^i = \lambda_k^o = 1$, $\beta = 15$ a $\nu = 0,001$. Velikost parametrů μ a ν ovlivňuje výslednou masku, při malých hodnotách se mohou vyskytnout redundantní regiony, naopak velké hodnoty mohou odstranit redundantní regiony, zároveň ale i části společných objektů. Počet iterací nastavili na $N_i = 1500$. Počáteční křivka Φ_k^0 je definována jako čtverec, jehož strany jsou blízko okrajům obrazu ve vzdálenosti $\gamma = 5$ pixelů, aby pokrýval většinu plochy společných objektů.

Objekt (y) definovaný vyhledávacím obrazem je detekován provedením kosegmentace nad každým extrahovaným rámcem videa. Schéma metody je zobrazeno na obr 2.19. Algoritmus metody je popsán v 1. Protože hodnota energetické funkce je počítána pro každý pixel, je počet iterací je spojen s počtem pixelů obrazu n . Výsledná časová složitost metody je úměrná k $O(n^2)$ [23].



Obrázek 2.21: Vývoj křivky, zde pět stavů odpovídajících iteracím 1, 101, 201, 301 a 401 [23].

Algorithm 1: METODA PRO DETEKCI OBJEKTŮ ZALOŽENÁ NA VYHLEDÁVACÍM OBRAZE

Input: Vyhledávací obraz **and** Video sekvence

- 1: **Initialization:** $\Phi_k^0 = 0$, $k = 0, 1$, $n = 0$ a ostatní parametry
 - 2: $temp_Phi_k^n \leftarrow \Phi_k^0$
 - 3: Extrahuj video rámeček $F_j \in \{f_1, f_2, \dots, f_t\}$
 - 4: **for** $j \leq t$ **do**
 - 5: **for** $n \leq maxiteration$ **do**
 - 6: Vypočti $r_\Phi(C_{1-k}^i)$ a $r_\Phi(C_k^o)$ pro každý obraz I_k
 - 7: Vypočti Φ_k^{n+1} řešením PDE v $temp_Phi_k^n$ za použití rovnice 2.19
 - 8: **if** $(\Phi_k^{n+1} = temp_Phi_k^n) \parallel (n = maxiteration)$ **then**
 - 9: **goto** Krok 16.
 - 10: **end**
 - 11: **else**
 - 12: $temp_Phi_k^n \leftarrow \Phi_k^{n+1}$
 - 13: **end**
 - 14: **end for**
 - 15: **end for**
 - 16: **Extrahuj společný objekt (y):** Pixely s hodnotou $\Phi_k^{n+1} = \Phi_k^n > 0$ náležející hledanému objektu (úm) pro F_j a Vyhledávací obraz.
 - 17: **Reinicializace:** $\Phi_k^0 = 0$, $k = 0, 1$, $n = 0$ a ostatní parametry
 - 18: $temp_Phi_k^n \leftarrow \Phi_k^0$
- Output:** Video sekvence s detekovaným objektem (y).
-

Kapitola 3

Návrh systému

Cílem této práce je navrhnout, implementovat a otestovat systém pro detekci a lokalizaci dronu či hejna dronů. V této kapitole bude popsán návrh systému. Popis implementace a testování systému se nachází v následujících kapitolách. Požadavky na systém jsou: systém bude pracovat v reálném čase, vstupem systému bude video zachycené pohyblivou kamerou nebo video výstup z pohybující se kamery, detekce dronu či hejna dronů bude co nejpřesnější. Výstupem systému bude video sekvence, kde v případě přítomnosti jednoho či více dronů budou tyto drony vyznačeny bounding boxem (co nejmenším obdélníkem ohraničujícím objekt).

3.1 Použitý software

Systém je navržen jako desktopová aplikace naprogramovaná v jazyce C++. Pro práci s obrazem bude využita knihovna OpenCV¹ (verze 4.1). Knihovna OpenCV je open source knihovna obsahující algoritmy pro zpracování obrazu a počítačové vidění. Algoritmus pro detekci bude založený na konvoluční neuronové síti. CNN model bude retrainován na Google Cloud ML Engine² pomocí platformy TensorFlow (verze 1.13.1)³, protože má širokou komunitu uživatelů, pokročilé nástroje a je k dispozici dostatek materiálů. Pro anotaci datasetu bude použit nástroj LabelImg (verze 1.8.1)⁴. Pro uživatelské rozhraní bude pro svoji univerzálnost a jednoduchost tvorby použita knihovna Qt⁵ (verze 5.11). Vývoj programu bude probíhat ve vývojovém prostředí Visual Studio 2017 (verze Community). Knihovny OpenCV a Qt jsou multiplatformní, aplikace tedy bude přeložitelná na různých systémech. V rámci této práce bude přeložena pro operační systém Microsoft Windows 10. Dokumentace zdrojových souborů bude vygenerována programem Doxygen⁶.

3.2 Návrh aplikace

Původní návrh počítal s použitím kosegmentace pro detekci objektů. Algoritmus je popsán v předchozí kapitole, konkrétně na práci pana Bandyopadhyaye [3]. Později jsem si uvědomil, že přes všechny výhody, které této algoritmus má je pro detekci dronů navzdory názvu

¹<https://opencv.org/>

²<https://cloud.google.com/ml-engine/>

³<https://www.tensorflow.org/>

⁴<https://github.com/tzutalin/labelImg>

⁵<https://www.qt.io/>

⁶<http://www.doxygen.nl/>

práce *Object of interest detection in video sequence using co-segmentation: A new era in video surveillance* naprosto nevhodný. Hlavním důvodem je extrémně vysoká výpočetní náročnost a tím i čas potřebný pro detekci dronu v jednom rámci videa. Algoritmus počítá energetickou funkci ve dvou obrazech pro každý pixel. I s nejmodernějším a nejrychlejším hardwarem a vysoce paralelizovanou a optimalizovanou implementací by výpočet jednoho rámce trval několik minut. Tento výrok potvrzuje navazující práce pana Soomra [31], kde jsou namísto pixelů použity superpixely a jejich počet je omezen na pouhých 1000. I přesto nejrychlejší výpočet trvá desítky sekund, což znemožňuje použití algoritmu pro detekci dronů, kde je vyžadován výpočet v reálném čase, případně v jednotkách sekund.

Z výše uvedených důvodů jsem návrh změnil. Namísto kosegmentace bude algoritmus pro detekci objektů založený na konvoluční neuronové síti, konkrétně na algoritmu Single Shot MultiBox Detector (SSD) [21]. Tento algoritmus představuje nejmodernější a nejpočetnější přístup pro daný problém. K tomuto účelu bude vytvořen a anotován zcela nový dataset obsahující různorodé obrázky dronů. Zdrojem dat pro dataset budou videozáběry poskytnuté vedoucím práce a videa a obrázky stažená z internetu, převážně z webu YouTube. Na internetu se mi i přes důkladné hledání nepodařilo najít jakýkoliv dataset obrázků dronů, rovněž zdroje na internetu byly překvapivě omezené. Vytvořený dataset tak bude pravděpodobně naprosto unikátní. Počítá se možností zveřejnění a zpřístupnění datasetu dalším výzkumníkům. Pro svou malou velikost v řádu tisíců příkladů bude vhodný jen pro retrainování předem natrénované sítě. Vstupem aplikace bude video sekvence čtená ze souboru nebo video výstup z kamery připojené přes rozhraní USB. Poté budou z videa extrahovány jednotlivé rámce. Další možnou úpravou pro urychlení výpočtu je vynechávání rámců, kde bude zpracován každý n-tý rámeček. Pomocí modulu dnn knihovny OpenCV bude realizován průchod sítí. Výstupem aplikace bude video sekvence s drony vyznačenými bounding boxy promítaná na obrazovku. Aplikace také bude umožňovat uložit výstup do souboru.

3.3 Uživatelské rozhraní

Okno aplikace bude rozděleno na dvě hlavní části: na levé straně bude video sekvence a na pravé ovládací prvky. Aplikace bude nabízet možnost výběru vstupu, nastavení parametrů algoritmu, spuštění nahrávání a cestu k souboru, možnost vypnutí/zapnutí zobrazení bounding boxu a spuštění/zastavení detekce.

Kapitola 4

Implementace systému

Kapitola 5

Testování

Kapitola 6

Závěr

Literatura

- [1] Achanta, R.; Shaji, A.; Smith, K.; aj.: SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE transactions on pattern analysis and machine intelligence*, ročník 34, 05 2012, doi:10.1109/TPAMI.2012.120.
- [2] Aker, C.; Kalkan, S.: Using Deep Networks for Drone Detection. 08 2017, doi:10.1109/AVSS.2017.8078539.
- [3] Bandyopadhyay, S.: Object of interest detection in video sequence using co-segmentation: A new era in video surveillance. 03 2018: s. 1–6, doi:10.1109/RAIT.2018.8389022.
- [4] Bay, H.; Ess, A.; Tuytelaars, T.; aj.: Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, ročník 110, 06 2008: s. 346–359, doi:10.1016/j.cviu.2007.09.014.
- [5] Chen, L.-C.; Papandreou, G.; Kokkinos, I.; aj.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, ročník 40, č. 4, 2018: s. 834–848.
- [6] Cheng, M.-M.; J Mitra, N.; Huang, X.; aj.: Salient Object Detection and Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ročník 37, 10 2011, doi:10.1109/TPAMI.2014.2345401.
- [7] Comaniciu, D.; Ramesh, V.; Meer, P.: Real-Time Tracking of Non-Rigid Objects using Mean Shift. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, ročník 2, 05 2000.
- [8] Cortes, C.; Vapnik, V.: Support Vector Network. *Machine Learning*, ročník 20, 09 1995: s. 273–297, doi:10.1007/BF00994018.
- [9] Dalal, N.; Triggs, B.: Histograms of Oriented Gradients for Human Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, ročník 2, 06 2005.
- [10] F. Chan, T.; Vese, L.: Active Contour Without Edges. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, ročník 10, 02 2001: s. 266–77, doi:10.1109/83.902291.
- [11] Freund, Y.; E Schapire, R.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. ročník 55, 12 1999: s. 119–139, doi:10.1006/jcss.1997.1504.

- [12] Geng, Q.; Zhou, Z.; Cao, X.: Survey of recent progress in semantic image segmentation with CNNs. *Science China Information Sciences*, ročník 61, č. 5, 2018: str. 051101.
- [13] Giraud, R.; Ta, V.; Papadakis, N.; aj.: Texture-Aware Superpixel Segmentation. 01 2019.
- [14] Guo, Z.: *Object detection and tracking in video*. [Online; navštíveno 23.01.2019]. URL <http://medianet.kent.edu/surveys/IAD01F-objdetection/index.html>
- [15] Han, S.; Shen, W.; Liu, Z.: *Deep Drone: Object Detection and Tracking for Smart Drones on Embedded System*. [Online; navštíveno 24.01.2019]. URL https://web.stanford.edu/class/cs231a/prev_projects_2016/deep-drone-object__2_.pdf
- [16] Horn, B.; G. Schunck, B.: Determining Optical Flow. *Artificial Intelligence*, ročník 17, 08 1981: s. 185–203, doi:10.1016/0004-3702(81)90024-2.
- [17] Jepson, D. F. . A.: *Markov Random Fields*. [Online; navštíveno 26.03.2019]. URL <http://www.cs.toronto.edu/~fleet/courses/2503/fall111/Handouts/mrf.pdf>
- [18] Joulin, A.; Bach, F.; Ponce, J.: Discriminative clustering for image co-segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 06 2010: s. 1943–1950, doi:10.1109/CVPR.2010.5539868.
- [19] Lee, Y.-B.; You, B.-J.; Lee, S.-W.: A real-time color-based object tracking robust to irregular illumination variations. ročník 2, 02 2001: s. 1659 – 1664 vol.2, doi:10.1109/ROBOT.2001.932849.
- [20] Li, S. Z.: *Markov Random Field Modeling in Computer Vision*. Berlin, Heidelberg: Springer-Verlag, 1995, ISBN 4-431-70145-1, [Online; navštíveno 26.03.2019]. URL http://www.nlpr.ia.ac.cn/users/szli/MRF_Book/MRF_Book.html
- [21] Liu, W.; Anguelov, D.; Erhan, D.; aj.: SSD: Single Shot MultiBox Detector. ročník 9905, 10 2016: s. 21–37, doi:10.1007/978-3-319-46448-0_2.
- [22] Liu, Z.; Wang, L.; Hua, G.; aj.: Joint Video Object Discovery and Segmentation by Coupled Dynamic Markov Networks. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, ročník 27, 07 2018, doi:10.1109/TIP.2018.2859622.
- [23] Meng, F.; Li, H.; Liu, G.; aj.: Image Cosegmentation by Incorporating Color Reward Strategy and Active Contour Model. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, ročník 43, 09 2012, doi:10.1109/TSMCB.2012.2215316.
- [24] Owczarek, M.; Strumillo, P.; Baranski, P.: Pedestrian tracking in video sequences: a particle filtering approach. 09 2015, doi:10.15439/2015F158.
- [25] Přemysl, K.: *Základy počítačové grafiky Studijní opora*. [Online; navštíveno 22.01.2019]. URL https://wis.fit.vutbr.cz/FIT/st/cfs.php?file=%2Fcourse%2FIZG-IT%2Ftexts%2Fizg_opora.pdf&cid=11496

- [26] R. Rodríguez-Canosa, G.; Thomas, S.; Cerro, J.; aj.: A Real-Time Method to Detect and Track Moving Objects (DATMO) from Unmanned Aerial Vehicles (UAVs) Using a Single Camera. *Remote Sensing*, vol. 4, issue 4, pp. 1090–1111, ročník 4, 04 2012: s. 1090–1111, doi:10.3390/rs4041090.
- [27] Rother, C.; Minka, T.; Blake, A.; aj.: Cosegmentation of Image Pairs by Histogram Matching - Incorporating a Global Constraint into MRFs. *CVPR*, ročník 1, 07 2006: s. 993– 1000, doi:10.1109/CVPR.2006.91.
- [28] Rozantsev, A.; Lepetit, V.; Fua, P.: Flying Objects Detection from a Single Moving Camera. 11 2014, doi:10.1109/CVPR.2015.7299040.
- [29] Shapiro Linda, G.: *Computer vision*. New Jersey: Prentice-Hall, vyd. 1 vydání, 2001, ISBN 0-13-030796-3.
- [30] Song, R.; Liu, M.; Wu, M.; aj.: A Shadow Elimination Algorithm Based on HSV Spatial Feature and Texture Feature. 05 2018: s. 585–591, doi:10.1007/978-3-319-59463-7_58.
- [31] Soomro, N.: An Efficient Image Co-segmentation Algorithm based on Active Contour and Image Saliency. 02 2016.
- [32] Svoboda, T.: *Image as Markov Random Field and Applications*. [Online; navštíveno 28.03.2019].
URL <http://cmp.felk.cvut.cz/~drbohlav/TeachPres/mrf.pdf>
- [33] Szeliski, R.: *Computer Vision : Algorithms and Applications*. London: Springer, 2010, ISBN 9781848829343.
- [34] Tani, T.; Sinha, S.; Sato, Y.: Joint Recovery of Dense Correspondence and Cosegmentation in Two Images. 06 2016: s. 4246–4255, doi:10.1109/CVPR.2016.460.
- [35] Tsai, Y.-H.; Yang, M.-H.; Black, M.: Video Segmentation via Object Flow. 06 2016: s. 3899–3908, doi:10.1109/CVPR.2016.423.
- [36] Tsai, Y.-H.; Zhong, G.; Yang, M.-H.: Semantic Co-segmentation in Videos. ročník 9908, 10 2016: s. 760–775, doi:10.1007/978-3-319-46493-0_46.
- [37] Wang, L.; Hua, G.; Sukthankar, R.; aj.: Video Object Discovery and Co-Segmentation with Extremely Weak Supervision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ročník PP, 10 2016: s. 1–1, doi:10.1109/TPAMI.2016.2612187.
- [38] Yazdi, M.; Bouwmans, T.: New Trends on Moving Object Detection in Video Images Captured by a moving Camera: A Survey. *Computer Science Review*, ročník 28, 05 2018: str. 157–177, doi:10.1016/j.cosrev.2018.03.001.
- [39] Zhang, H.-Y.: Multiple moving objects detection and tracking based on optical flow in polar-log images. *2010 International Conference on Machine Learning and Cybernetics, ICMLC 2010*, ročník 3, 07 2010: s. 1577–1582, doi:10.1109/ICMLC.2010.5580801.
- [40] Zhao, H.; Qi, X.; Shen, X.; aj.: Icnnet for real-time semantic segmentation on high-resolution images. 2018: s. 405–420.