

# DETECTION ALGORITHMS OF MOVING OBJECTS

**Josef Novotný**

Bachelor Programme (3), FEEC BUT

E-mail: xnovot0v@vutbr.cz

**Petr Marcoň**

E-mail: marcon@feec.vutbr.cz

**Abstract:** The aim of this work is to design automatic detection system on Raspberry Pi. Device includes a camera with a swivel module, which can set a proper angle. It involves applying an algorithm based on objects detection and recognition methods. Algorithms have been chosen according to real-time implementation and allows to send a warning when detecting unwanted objects. Algorithm was verified on test videos of drones and landing airplanes.

**Keywords:** Object detection, Object Recognition, Raspberry Pi, OpenCV

## 1. ÚVOD

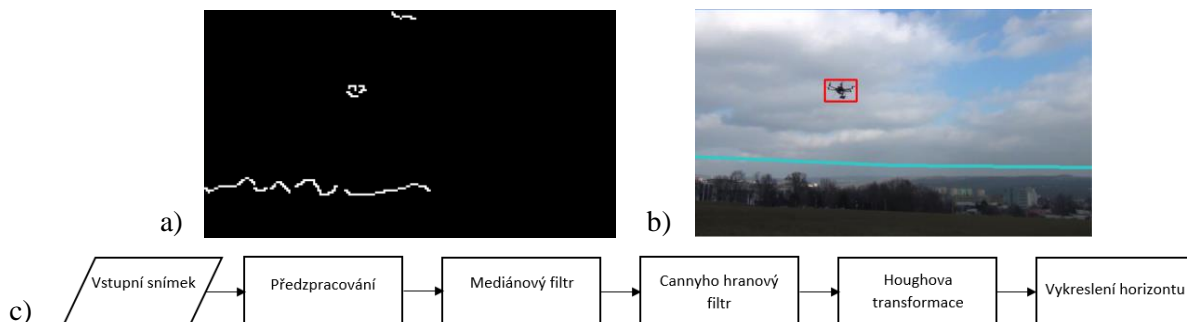
Automatický detekční systém nachází v současnosti řadu uplatnění, například v okolí letišť či střežených areálů. V případě zachycení nežádoucího objektu je snímek uložen a spolu s ním je zasláno upozornění. Umožňuje tak vizuální kontrolu nad vymezeným prostorem. Vytvořený algoritmus rozeznává horizont, a to i v případě pohybu kamery. Otáčecí modul pak zajišťuje natočení kamery, které může probíhat automaticky za detekovaným objektem. Projekt je implementovaný na Raspberry Pi, které bylo v rámci testování ověřeno jako výkonově dostačující.

## 2. PŘEDZPRACOVÁNÍ OBRAZU

V prvním kroku jsou načteny vstupní data, a to buď přímo z kamery nebo ze záznamu. Dále dochází ke zmenšení rozlišení obrazu na šířku 600 pixelů (výška závisí na poměru stran snímků). Poté je z obrazu odstraněna barevná složka. Veškeré další zpracování probíhá pomocí knihovny pro zpracování obrazu OpenCV.

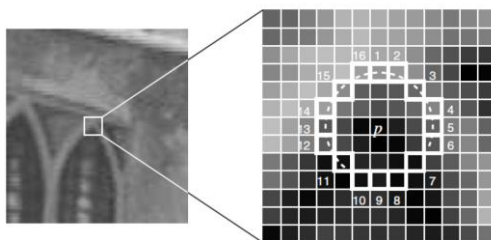
## 3. DETEKCE HORIZONTU A OBJEKTŮ

Po fázi předzpracování obrazu je ve snímcích za pomoci Cannyho hranového filtru a Houghovy transformace detekován horizont (obrázek 1c). Výstupem hranového filtru je černobílý obraz (obrázek 1a), který indikuje hranové pixely, ve kterých Houghova transformace nalezne nejvýraznější křivky. Tyto křivky dávají dohromady lomenou čáru, která je brána jako horizont (obrázek 1b). Jestliže je známa poloha horizontu, pak jsou detekovány všechny objekty nad touto křivkou.



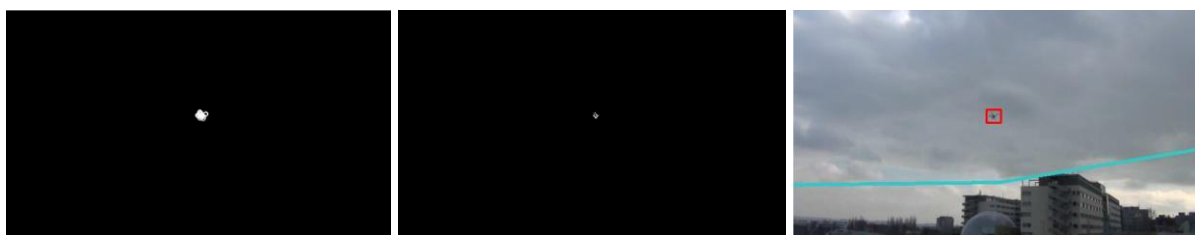
**Obrázek 1:** Detekce horizontu po aplikaci Cannyho hranového filtru a Houghovy transformace

Pro detekci objektů na obloze byla použita metoda ORB (z angl. Oriented FAST and Rotated BRIEF). Ta využívá detektoru významných bodů FAST (z angl. Features from accelerated segment test). Detektor FAST je založen na principu segmentového testu, který uvažuje kruhovou oblast šestnácti pixelů kolem hledaného rohu  $p$  (viz obrázek 2). Rozhodující pro tento test je podmínka nalezení 12 kontinuálních pixelů v kruhové oblasti, které jsou světlejší než  $I(p)+t$ , anebo jestliže jsou všechny body tmavší než  $I(p)$ .  $I(p)$  je intenzita v bodě  $p$  a  $t$  je zvolený práh. [1] Splněním této podmínky je zajištěno, že jsou detekovány vždy pixely, který jsou kontrastní od okolí.



**Obrázek 2:** Lokalizace bodů ORB pomocí segmentového testu. [2]

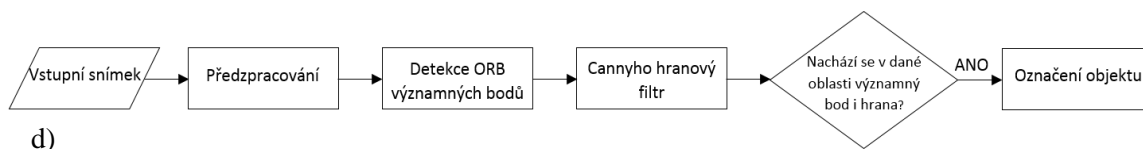
Množina všech nalezených bodů ORB je vykreslena na čistou masku (obrázek 3a). Tato oblast je zároveň testována na přítomnost hrany (obrázek 3b). Pokud jsou obě podmínky splněny, je oblast s objektem vyznačena (obrázek 3c). V opačném případě, kdy hrana není nalezena, je oblast vyřazena z detekce. Tímto způsobem lze předcházet chybné detekci, která by mohla nastat při větší oblačnosti.



a)

b)

c)



d)

**Obrázek 3:** Masky významných bodů ORB a), obraz po použití Cannyho hranového filtru b), detekovaný objekt a horizont c), vývojový diagram d)

## 4. ROZPOZNÁVÁNÍ OBJEKTŮ

V případě nasazení detektoru v blízkosti letišť je možné rozpoznat přistávající letadla od okolí. Pro jejich rozlišení byla použita metoda založená na Viola-Jones detektoru s využitím Haarových příznaků.

### 4.1. HAAROVY PŘÍZNAKY

Metoda Viola-Jones je výhodná vzhledem k výpočetní rychlosti. Tento způsob detekce je velmi spolehlivý a umožňuje tak využití i v aplikacích pracujících v reálném čase. Implementace metody se sestává ze tří dílčích kroků. V prvním proběhne výpočet integrálního obrazu s využitím obdélníkových Haarových příznaků. Dalším krokem je učící algoritmus AdaBoost. Ten vybírá určitý malý počet vizuálních příznaků z většího počtu snímků (obrázek 4a) s jejichž pomocí sestaví klasifikátor. V poslední řadě se tyto klasifikátory sdružují do kaskád, čímž se dosáhne větší efektivity klasifikátoru. K účelu trénování je zapotřebí velkého množství snímků, jak pozitivních, tak i negativních.

Samotný algoritmus vygeneruje klasifikátor, který je uložen jako XML soubor a definuje pozice a parametry jednotlivých obrazových příznaků, které je potřeba vyhodnotit pro výpočet odezvy klasifikátoru. [3] Výsledek detekce je vyznačen zeleným obdélníkem jako rozpoznáný objekt (obrázek 4b).



**Obrázek 4:** Příklad trénovací sady snímků a), snímek s rozpoznaným objektem b)

## 5. ZÁVĚR

Cílem práce bylo najít vhodný způsob pro detekci pohybujících se objektů. Toho bylo dosaženo pomocí významných bodů ORB testovaných spolu s Cannyho hranovým detektorem. Projekt je dále doplněn o rozlišení nežádoucích objektů, jako jsou ptáci či drony, od přistávajících letadel. Výhodou algoritmu je možnost dynamicky pohybovat kamerou. V rámci testování bylo zjištěno, že detekční algoritmy vyhodnocují objekt bezchybně pouze na bezmračné obloze. Nejvíce chyb pak vzniká při kupovité oblačnosti, kdy algoritmus v řadě případů chybně detekuje hrany mraků jako objekty. Velkou část chyb lze odstranit nastavením minimální velikosti zkoumaného objektu. Náhodné chyby detekce pak lze odstranit softwarovým algoritmem.

Mikropočítač Raspberry Pi byl zvolen vzhledem k předpokládanému dostatečnému výkonu. Ve stejné kategorii lze použít i Orange Pi, či jiné výkonnější zařízení, avšak pro účely detekce objektů je Raspberry Pi zcela postačující. Výhodou Raspberry Pi je především rozsáhlá dokumentace ke knihovně OpenCV a možnost o doplnění kompatibilním otočným modulem s kamerou. V neposlední řadě jde o jednu z nejrozšířenějších open-sourceových platform mikropočítačů. Přesto bylo nutné kód maximálně optimalizovat, aby dílčí podprocesy byly zpracovány v dostatečné rychlosti (uvedeno v tab.1). Jelikož je program rozvětvený, jsou jednotlivé části závislé na tom, zda právě detekují objekt či více objektů. Výsledkem je zpracování dvou až tří snímků za sekundu.

Process	Minimum	Maximum	Průměr
Předzpracování	116 ms	200 ms	150 ms
Detekce horizontu	27 ms	119 ms	50 ms
Detekce objektů	151 ms	250 ms	200 ms
Rozpoznávání objektů	52 ms	55 ms	55 ms
<b>Celkem</b>	<b>348 ms</b>	<b>624 ms</b>	<b>455 ms</b>

**Tabulka 1:** Přehled výpočetní náročnosti pro jednotlivé procesy

## REFERENCE

- [1] XIE, Songyun, Wanpeng ZHANG, Wang YING a Khalid ZAKIM: Fast Detecting Moving Objects in Moving Background using ORB: Feature Matching [online]. Beijing, 2013, s. 2
- [2] ROSTEN, Edward: FAST Corner Detection, 2015, Dostupné z: <https://www.edwardrosten.com/work/fast.html>
- [3] CEN, Kaiqi: Study of Viola-Jones Real Time Face Detector, Dostupné z: [https://web.stanford.edu/class/cs231a/prev\\_projects\\_2016/cs231a\\_final\\_report.pdf](https://web.stanford.edu/class/cs231a/prev_projects_2016/cs231a_final_report.pdf)