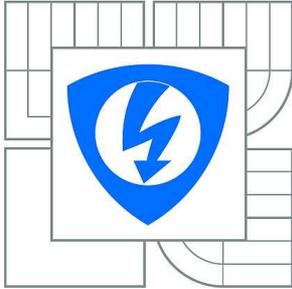


**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

## **STABILIZACE OBRAZU**

IMAGE STABILIZATION

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

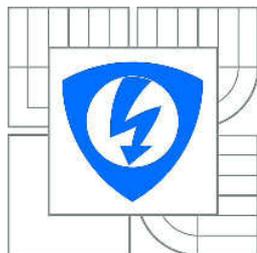
**Bc. MAREK OHRÁDKA**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. PETR ČÍKA, Ph.D.**

BRNO 2012



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Diplomová práce

magisterský navazující studijní obor  
Telekomunikační a informační technika

**Student:** Bc. Marek Ohrádka  
**Ročník:** 2

**ID:** 78824  
**Akademický rok:** 2011/2012

## NÁZEV TÉMATU:

**Stabilizace obrazu**

## POKYNY PRO VYPRACOVÁNÍ:

Prostudujte možnosti automatické stabilizace obrazu. Vytvořte si vlastní sekvenci snímků, kde bude jeden snímán jeden obrázek, který bude v určité míře geometricky transformován tak, aby byl patrný třes. Navrhněte metodu, která video na této sekvenci stabilizuje. Tuto metodu implementujte do prostředí programu RapidMiner.

## DOPORUČENÁ LITERATURA:

[1] BURGER, Wilhelm; BURGE, Mark J. Principles of Digital Image Processing: Fundamental Techniques. Londýn : Springer, 2009. 272 s. ISBN 978-1848001909.

[2] GONZALEZ, Rafael C.; WOODS, Richard E. Digital Image Processing. 3. Londýn : Pearson Pentice Hall, 2008. 954 s. ISBN 978-0-13-505267-9.

**Termín zadání:** 6.2.2012

**Termín odevzdání:** 24.5.2012

**Vedoucí práce:** Ing. Petr Číka, Ph.D.

**Konzultanti diplomové práce:**

**prof. Ing. Kamil Vrba, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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.

## ANOTACE

Tato práce se zabývá tématem digitální stabilizace obrazu. Obsahuje stručný přehled problematiky a dostupných metod. Cílem bylo navrhnout a implementovat systém stabilizace obrazu v jazyku JAVA, který bude určený pro použití v prostředí RapidMiner. Byly navrženy dvě metody stabilizace. První je založena na principu odhadu a kompenzace pohybu s využitím blokových vyhledávacích algoritmů Full-search a Three-step search. Základem druhé metody je detekce hranice objektu. Funkčnost navržené metody byla testována na vytvořených videosekvencích, ve kterých je viditelný třes. Testování ukázalo, že při vhodném nastavení vstupních parametrů metody založené na detekci hranice objektu je dosahováno úspěšné stabilizace scény. Míra snížení chyby mezi snímky dosahuje přibližně o 65 – 85 %. Výstupem metody je stabilizovaná sekvence snímků a soubor metadat získaných v průběhu stabilizace, které mohou být dále zpracovány v prostředí RapidMiner.

**Klíčová slova:** digitální stabilizace obrazu, videosekvence, Rapidminer, Image Miner, IMMI, Image Sequence Stabilizer, odhad pohybu, kompenzace pohybu, vektor pohybu, normalizace obrazu, prahování obrazu, detekce hranice objektu, MAE, SAE, vyplňování okrajů snímků

## ABSTRACT

This thesis deals with digital image stabilization. It contains a brief overview of the problem and available methods for digital image stabilization. The aim was to design and implement image stabilization system in JAVA, which is designed for RapidMiner. Two new stabilization methods have been proposed. The first is based on the motion estimation and motion compensation using Full-search and Three-step search algorithms. The basis of the second method is the detection of object boundaries. The functionality of the proposed method was tested on video sequences with contain visible shake of the scene, which has been created for this purpose. Testing results show that with the proper set of input parameters for the object border detection method, successful stabilization of the scene is achieved. The rate of error reduction between images is approximately about 65 to 85%. The output of the method is stabilized image sequence and a set of metadata collected during stabilization, which can be further processed in an environment of RapidMiner.

**Keywords:** digital image stabilization, video sequences, Rapidminer, Image Miner, IMMI, motion estimation, motion compensation, motion vector, Image Sequence Stabilizer, image normalization, image thresholding, object border detection, MAE, SAE, image border reconstruction

**OHRÁDKA, M.** *Stabilizace obrazu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 42 s. Vedoucí diplomové práce Ing. Petr Číka, Ph.D..

# PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Stabilizace obrazu“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne 23.5.2012

.....

podpis autora

# POĎAKOVANIE

Ďakujem vedúcemu svojej diplomovej práce Ing. Petrovi Číkovi, Ph.D. za jeho trpezlivosť a takisto svojim rodičom a priateľke za ich nekonečnú podporu.

V Brně dne 23.5.2012

.....

podpis autora

Výzkum popsany v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

# Obsah

1 Úvod.....	3
2 Historické pozadie problému.....	4
2.1 Vznik chvenia.....	4
3 Techniky stabilizácie obrazu.....	5
3.1 Mechanické prostriedky.....	5
3.2 Optická stabilizácia.....	8
3.3 Digitálna stabilizácia [3].....	10
3.4 Elektronická stabilizácia.....	10
4 Kompresné algoritmy – základ stabilizácie obrazu.....	12
5 Návrh metódy pre stabilizáciu obrazu.....	15
5.1 Metóda stabilizácie založená na blokovom vyhľadávaní.....	15
5.2 Metóda stabilizácie založená na detekcii hranice objektu.....	17
6 RapidMiner a rozšírenie Image Miner.....	20
6.1 Predstavenie rozšírenia Image Miner (IMMI).....	21
7 Praktická realizácia operátora stabilizácie obrazu.....	23
7.1 Princíp funkčnosti operátora.....	24
7.2 Popis parametrov operátora Image Sequence Stabilizer.....	25
7.3 Prevod do stupňov šedi a normalizácia.....	27
7.4 Určenie rozsahu a práca s oblasťou ROI.....	28
7.5 Implementované metódy vyhľadávania vektoru pohybu.....	28
7.5.1 Full-Search.....	29
7.5.2 Three-step search.....	30
7.5.3 Object border detection.....	31
7.6 Vykresľovanie vektoru pohybu a oblasti ROI.....	32
7.7 Vypĺňanie oblastí bez platnej obrazovej informácie.....	33
7.8 Výpočet hodnoty MAE pre upravené snímky.....	33
8 Testovanie a výsledky.....	35
8.1 Videosekvencie.....	35
8.2 Dosiahnuté výsledky.....	36
8.3 Nastavovanie hodnoty prahu.....	37
8.4 Rozsah riadkov a stĺpcov.....	39
8.5 Výkonnostné testy.....	39
9 Záver.....	40
10 Zoznam použitej literatúry.....	41

## Zoznam obrázkov

Obrázok 3.1: Proaim Shoulder Mount. ....	6
Obrázok 3.2: Použitie Steadicam v praxi. ....	6
Obrázok 3.3: Kenyon Gyro Stabilizer KS-6. ....	7
Obrázok 3.4: Koľajnicový systém RailCam od spoločnosti Camera support. ....	7
Obrázok 3.5: Teleskopické kamerové rameno založené na systéme Techno 15 Technocrane od firmy Supertechno. ....	8
Obrázok 3.6: Stabilizačný systém Nikon VR. ....	9
Obrázok 3.7: Konica Minolta - posuv snímacieho čipu. ....	9
Obrázok 3.8: Schéma algoritmu odstránenia rozmazania snímku. ....	11
Obrázok 4.1: Ukážka dvoch po sebe nasledujúcich snímkov videa. ....	12
Obrázok 4.2: Znázornenie vplyvu použitia kompenzácie pohybu na obsah rozdielového snímku. ....	13
Obrázok 4.3: Vektory pohybu vykreslené v rekonštruovanom snímku. ....	14
Obrázok 5.1: Princíp vyhľadávania vektoru pohybu. ....	16
Obrázok 5.2: Ukážka kompenzácie pohybu; ....	16
Obrázok 5.3: Princíp detekcie hranice objektu ....	17
Obrázok 6.1: Ukážka programového prostredia RapidMiner s návrhom procesu segmentácie obrazu. ....	20
Obrázok 6.2: Prehľad hlavných kategórií operátorov rozšírenia IMMI; čísla v zátvorkách uvádzajú počet operátorov v danej kategórii. ....	22
Obrázok 7.1: Ukážka užívateľského rozhrania operátora Image Sequence Stabilizer a jeho parametrov v prostredí RapidMiner. ....	23
Obrázok 7.2: Princíp prehľadávania algoritmu Three-step search. ....	30
Obrázok 7.3: Ukážka vykresľovania vektoru pohybu a oblasti ROI. ....	32
Obrázok 7.4: Ukážka vyplňania okrajových oblastí snímky bez platných obrazových dát ....	33
Obrázok 8.1: Ukážka vplyvu nastavenej hodnoty prahu na výstupný snímok ....	38

## 1 Úvod

S postupujúcim vývojom technológií v oblasti zachytávania a spracovávania obrazu sa čoraz častejšie, alebo snáď už výhradne, používa digitálne spracovanie obrazu. Analógové spracovanie je spojené s viacerými nevýhodami, akými sú napríklad cena, čas a nepohodlné použitie a preto je v dnešnej dobe využívané prevažne už len v oblasti umeleckej tvorby. Digitalizácia tohoto procesu so sebou priniesla nespočetné výhody, ako napríklad možnosť postprodukcie, retušovania a nepopierateľne jednoduchšieho skladovania obrazového materiálu. Jedným z problémov, ktorý sa v tejto oblasti riešil a stále rieši je stabilizácia obrazu a práve tomu je venovaná táto práca. Zaujíma vás technologické pozadie tohoto problému, príčina jeho vzniku, potreba jeho riešenia a možnosti, ktorými je možné ho dosiahnuť? V nasledujúcich kapitolách sú spracované a vysvetlené viaceré otázky súvisiace s týmto problémom a jeho riešením vrátane návrhu funkčného systému stabilizácie obrazu.

Cieľom tejto práce je vytvorenie systému pre digitálnu stabilizáciu obrazu aplikovateľného na sekvenciu snímok, ktorý bude spracovaný v jazyku JAVA pre použitie v prostredí RapidMiner. Takisto budú vytvorené videosekvencie, na ktorých bude demonštrovaná funkčnosť navrhnutého systému.

## 2 *Historické pozadie problému*

Počas doby používania fotografických prístrojov vývoj stále smeroval k zmenšovaniu a zjednodušovaniu technického vybavenia a zlepšovaniu kvality záznamových médií. Cieľom bolo, a aj stále je, zachytávať kvalitné snímky, čo je niekedy v závislosti na svetelné podmienky veľmi problematické. Na statických snímkach sa prejavuje plynutie času rozmazaním snímky. To vzniká jednak pohybom snímaných objektov a jednak pohybom samotného snímacieho zariadenia.

S príchodom filmového priemyslu sa začal tento problém zameriavať hlavne na pohyb snímacieho zariadenia. Tento nežiadúci pohyb, hlavne chvenie, sa totiž prejavuje viditeľne na zaznamenávanej sekvencii obrazu a pri jej prehrávaní pôsobí väčšinou rušivo. V profesionálnej sfére sa preto začali používať rôzne prostriedky, ktoré majú za úlohu chvenie eliminovať.

### 2.1 *Vznik chvenia*

Drvivá väčšina chvenia vzniká pohybmi ľudského tela či už vedome alebo nevedome. Bez nášho vedomia sa naše telo neustále musí vysporiadavať s udržiavaním rovnováhy, čo je zabezpečované aktiváciou rôznych skupín svalov nášho tela. Vo výsledku sa to pri nakrúcaní videa prejavuje tým, že sa kamera držaná v ruke pohybuje vo vodorovnom aj zvislom smere. Keďže optická sústava kamery prenáša obraz scény na snímací čip, ktorý ma miniatúrne rozmery<sup>1</sup>, znamená to, že aj relatívne malý pohyb kamery môže spôsobiť posun scény v obraze o výrazne badateľnú odchýlku. Keď sa k tomu pridá chvenie ľudského tela, spôsobuje to zvyčajne roztrasený alebo rozmazaný záznam. Ktorý je pri reprodukovani náročný na sústredenie.

---

1 Veľkosť čipu APS-C je približne 25,1 x 16,7 mm

### 3 *Techniky stabilizácie obrazu*

Pri obstarávaní záznamu scény je možné použiť niekoľko základných techník.

Možno ich rozdeliť do týchto kategórií:

- mechanické
- optické
- elektronické
- digitálne

Prvá kategória zjednocuje prostriedky a techniky, pre ktoré je spoločným znakom použitie určitého externého vybavenia spojeného s kamerou alebo fotoaparátom. Ostatné techniky sú implementované priamo v tele snímacieho zariadenia, v snahe minimalizovať rozmery a hmotnosť kompletného technického vybavenia pri zachovaní kvality záznamu.

#### 3.1 *Mechanické prostriedky*

Prvým prostriedkom, ktorý sa používal už od počiatkov je **statív**. Jeho použitie bolo nutnosťou, pretože už samotná hmotnosť fotografického alebo filmového vybavenia prakticky vylučovala možnosť získať snímku alebo záznam z ruky. Používanie statívu však znemožňuje priestorovú dynamiku pri práci so záberom, a preto nie je vždy vhodné. Navyše so spoľahlivosťou a pevnosťou statívu narastá väčšinou aj jeho hmotnosť.

Najjednoduchšou technikou, ktorá je používaná pri zaznamenávaní filmových sekvencií, je umiestnenie kamery **na rameno** kameramana, čo zaistí relatívne stabilnú oporu pre kameru. Zábery získané touto technikou majú však stále svoje nedostatky, ktoré vznikajú pri pohyboch tela kameramana, najmä pri jeho chôdzi. S postupujúcim pokrokom techniky a minimalizáciou elektronických zariadení však táto technika prestala byť vhodná pre nové kompaktné kamery, a preto boli vyvinuté

pomocné podporné konštrukcie (viz obr. 3.1), ktoré umožňujú inštaláciu aj takýchto malých zariadení a následnú prácu s nimi v pohodlnej polohe.

Ďalší pomocný podporný systém, ktorý využíva možnosti dynamiky ľudského tela je systém *Steadicam*, viz obrázok 3.2. Pozostáva z postroja, ktorý je upevnený na tele kameramana a z odpruženého ramena so závažiami, na ktorom je pripevnená samotná kamera.



Obrázok 3.1: Proaim Shoulder Mount.  
Zdroj: <http://www.thecinecity.com/eshop/images/T/c.support-8.jpg>



Obrázok 3.2: Použitie Steadicam v praxi.  
Zdroj: [http://www.creativevideo.co.uk/images/products/altimage/22%2009%202006steadicam\\_flyer\\_1.jpg](http://www.creativevideo.co.uk/images/products/altimage/22%2009%202006steadicam_flyer_1.jpg)

V situáciách, kedy nie je možné použiť žiaden z vyššie uvedených prostriedkov bola vyvinutá technika stabilizácie na báze **gyroskopu**. Pomocou fixného spojenia tela kamery s aktívnym gyroskopom je dosiahnuté odrušenie vplyvu vonkajších vibrácií na kameru. Príkladom takéhoto systému je *Kenyon Gyro Stabilizer KS-6*, viz obrázok 3.3. Táto technika je často používaná pri získavaní záberov z pohybujúcich sa objektov, akým je napríklad terénne auto, lietadlo alebo helikoptéra, pričom je kameraman priestorovo značne limitovaný vnútornými rozmermi konkrétneho vozidla. Gyroskop spoľahlivo eliminuje vibrácie motora a udržiava kameru v prirodzenej polohe vzhľadom k snímanej scéne. Orientačná cena takéhoto systému sa pohybuje v rozmedzí od 1000 do 9000 USD [1] v závislosti od potrebnej veľkosti a vybavenia modelu.



Obrázok 3.3: Kenyon Gyro Stabilizer KS-6.

Zdroj: <http://www.nikon-fotografie.de/vbulletin/nikon-d3-fx/66221-nikon-d3-plus-nikkor-af-s-500mm-f-4-0-und-dem-ks-6-kenyon-labs-gyro-freihand.html>

V rámci štúdiovej techniky sa často používajú **koľajnicové systémy**, ktoré zaisťujú stabilitu snímanej scény pri súčasnom zachovaní priestorovej dynamiky pohybu kamery. Takýmto systémom je napríklad *RailCam* od spoločnosti *Camera support*. Pohyb kamery je ale v tomto prípade obmedzený na horizontálny smer.



Obrázok 3.4: Koľajnicový systém RailCam od spoločnosti Camera support.

Zdroj: <http://www.camerasupport.com/RailCam/railcam.html>

Ďalej sú používané **pohyblivé platformy** pre kamery na báze výsuvného ramena, ktoré umožňujú pohyb ako v horizontálnej, tak vo vertikálnej rovine.



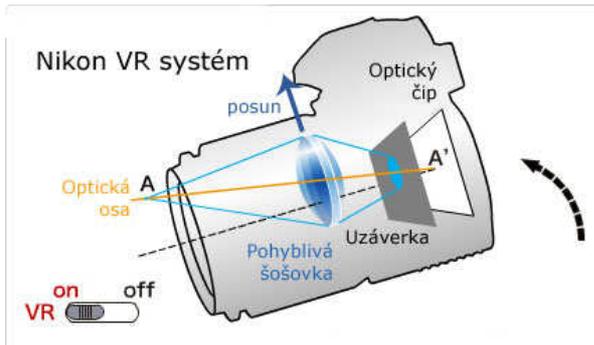
Obrázok 3.5: Teleskopické kamerové rameno založené na systéme Techno 15 Technocrane od firmy Supertechno.

Zdroj: <http://pacificmotion.net/WP2/>

Posledné dva systémy sú použiteľné len v profesionálnej sfére nakoľko vyžadujú zložitú montáž a obsluhu, nehovoriac o finančných nákladoch potrebných na zaobstaranie a údržbu. Z týchto dôvodov bol výskum zameraný na alternatívne metódy zabezpečenia stability obrazu.

### 3.2 Optická stabilizácia

V súčasnej dobe je v komerčnom sektore za najúčinnjšiu metódu považovaná optická stabilizácia obrazu. Táto technika je založená na prítomnosti pohyblivého člena v optickej sústave prístroja, ktorý zmenou svojej polohy spôsobí zmenu optickej cesty k senzoru. Zmena polohy je riadená mikroprocesorom, ktorý vyhodnocuje informácie získané z pohybových senzorov vnútri prístroja.



Obrázok 3.6: Stabilizačný systém Nikon VR.  
Zdroj: [http://www.nikon.com/about/technology/core/software/vr\\_el/index.htm](http://www.nikon.com/about/technology/core/software/vr_el/index.htm)



Obrázok 3.7: Konica Minolta - posuv snímacieho čipu.  
Zdroj: [2]

Existujú dve známe možnosti implementácie, ktorými možno dosiahnuť elimináciu vibrácií:

- použitím pohyblivého optického člena **v objektíve** (viz obr. 3.6)
- **zabudovaním** pohyblivého snímacieho čipu **do tela samotného prístroja** (viz obr. 3.7)

Prvú techniku používajú napríklad firmy *Nikon*, *Canon* a *Panasonic*, pričom firma *Canon* drží prvenstvo v predstavení tohoto systému na trhu v roku 1995 [2]. Druhý spôsob stabilizácie je používaný v zariadeniach *Konica Minolta*, *Olympus*, *Sony*, *Pentax* a v ďalších.

Výhodou systému zabudovaného v tele prístroja je jeho nižšia obstarávacia cena a možnosť použitia v kombinácii prakticky s akýmkoľvek objektívom. Na druhej strane však systém implementovaný priamo v objektíve vykazuje lepšiu kvalitu snímok pri dlhších ohniskových vzdialenostiach používaných v teleobjektívoch. Ďalšou nespornou výhodou je fakt, že obraz je stabilizovaný už aj v optickom hľadáčku, čo zvyšuje užívateľský komfort už pri kompozícii scény, a toto pri použití systému s pohybom čipu nie je možné. Nevýhodou však ostáva nutnosť opakovane platiť za tento systém pri kúpe každého nového objektívu.

Oba tieto systémy úspešne kompenzujú chvenie obrazu spôsobené chvením rúk fotografa avšak nie sú určené pre odstránenie pohybového rozmazania objektu spôsobené pohybom samotného objektu.

### 3.3 Digitálna stabilizácia [3]

Digitálna stabilizácia obrazu je založená na softwarovom riešení problému, čím sa značne znižujú náklady spojené s touto technológiou. Navyše nie je potrebné žiadne špeciálne hardwarové zariadenie, a preto je možné spomínanú metódu dodatočne doplniť aj do starších modelov kamier pomocou aktualizácie firmwaru, alebo ju aplikovať vo fáze postprodukcie na záznam získaný bez stabilizácie obrazu.

Princíp fungovania je založený na detekcii rozdielov dvoch po sebe nasledujúcich snímok v sekvencii. Podľa posunu väčšiny blokov v obraze, ktoré štandardne tvoria pozadie scény, je vypočítaný vektor posunutia obrazu. Tento vektor je použitý v nasledujúcom snímku pre určenie posunutia kompenzovaného stredu scény. Výsledný obraz scény nie je tvorený celou plochou snímacieho čipu, ale jej výrezom. Okrajové body mimo výrezu sú použité ako vyrovnávací pamäť pri kompenzácii pohybu scény aby nedochádzalo k zobrazovaniu oblastí bez obrazových dát (vo videozázname sa prejavujú ako čierne pruhy). Pri použití tejto techniky je upravené rozlíšenie výslednej scény.

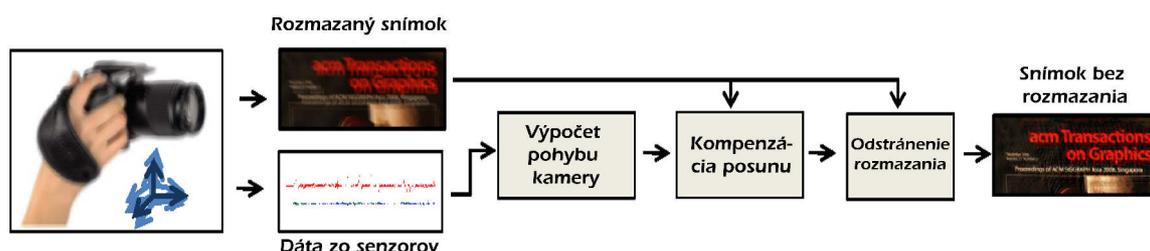
Podobné výsledky je možné dosiahnuť aj bez použitia výrezu obrazu. Je však nutné v okrajových častiach posunutých snímok doplniť chýbajúcu obrazovú informáciu. To je riešené extrapoláciou dát z predchádzajúceho a aktuálneho snímku.

Zásadným problémom, ktorý je pri tejto metóde potrebné riešiť je správna klasifikácia žiadúcich a nežiadúcich pohybov v scéne, na základe ktorých sa vypočítava vektor posunutia. V určitých prípadoch by tak mohol byť nesprávne kompenzovaný prirodzený pohyb objektu v scéne, pričom by telo kamery mohlo byť počas záznamu bez pohybu.

### 3.4 Elektronická stabilizácia

Elektronická stabilizácia je v podstate spojením vlastností predchádzajúcich dvoch metód. Z pohybových snímačov v tele kamery sú získavané informácie

o pohyboch a vibráciách, na základe ktorých je softwarovo korigovaná pozícia výrezu oblasti na obrazovom čipe. Opäť hrá v tomto prípade kľúčovú úlohu fakt, že vo výslednom snímku nie je využitá celá plocha snímacieho čipu, ale len jej časť, pričom sú okrajové body využité ako vyrovnávacia pamäť. Laboratórny prípravok skonštruovaný za účelom stabilizácie obrazu bol vytvorený a testovaný skupinou *Microsoft Research*. Výsledky sú spracované v [4].



Obrázok 3.8: Schéma algoritmu odstránenia rozmazania snímku.

Zdroj: [4]

Použitím senzorov pohybu odpadá nutnosť riešiť problém s klasifikáciou pohybov v scéne, ktoré vznikli chvením kamery a tých, ktoré boli spôsobené pohybom samotných snímaných objektov. Sensory totiž správne zaznamenajú len pohyby tela prístroja.

Cenovo je použitie tejto metódy nenáročné, čo ho predurčuje pre použitie v kompaktných prístrojoch strednej kategórie určených pre širokú skupinu užívateľov.

## 4 Kompresné algoritmy – základ stabilizácie obrazu

Príchod digitálnych videozáznamov priniesol so sebou množstvo výhod. Medzi hlavné z nich patria nasledovné:

- digitalizovaný signál možno uložiť, kopírovať a opakovane prehrávať bez straty kvality,
- do prenášaného videa nie je počas prenosu pridávaný žiaden šum,
- digitálne video je možné jednoducho prispôbovať a upravovať.

Problémom digitálneho videa je jeho veľkosť. Každé video je snímané a ukladané vo forme sekvencie snímok. Dôvodom je prirodzená vlastnosť ľudského oka, ktoré pri dostatočne vysokej frekvencii zobrazovania nie je schopné vnímať prechody medzi jednotlivými snímkami. Výsledkom je ilúzia súvislého pohybu objektov v scéne.

Štandardne sa ukladá 24 – 30 snímok na 1 sekundu záznamu [5]. V nekomprimovanej podobe by jedna sekunda video signálu štandardného televízneho formátu zaberala cca 32 MB, film s dĺžkou 90 minút analogicky zaberie cca 170 GB dát. Toto je hlavným dôvodom, pre ktorý je potrebné redukovať veľkosť digitálneho videa. Pre tento účel sa používajú kompresné štandardy, z ktorých najviac používané sú vyvíjané organizáciami ITU-T (H.261, H.263, H.264) a ISO/IEC (MPEG-1, MPEG-2, MPEG-4).



Obrázok 4.1: Ukážka dvoch po sebe nasledujúcich snímkov videa.

Zdroj: [http://compression.ru/video/motion\\_estimation/index\\_en.html](http://compression.ru/video/motion_estimation/index_en.html)

Základom pri kódovaní videa je fakt, že snímky, ktoré nasledujú tesne po sebe sa väčšinou od seba líšia len minimálnymi detailami. Preto je možné použiť predikciu snímky zo snímky predchádzajúcej. Úspora množstva dát je teda docielená tým, že sa ukladajú len rozdiely obrazových dát medzi dvoma po sebe nasledujúcimi snímkami (viz obr. 4.1 a 4.2).

Ďalšou dôležitou súčasťou týchto štandardov je aj použitie niektorej z metód kompenzácie pohybu. S jej použitím je možné ušetriť ešte väčšie množstvo prenášaných dát. Pri zachovaní rovnakej kvality výstupu pri kompresii sú metódy bez kompenzácie pohybu limitované kompresným pomerom 5:1, pričom s použitím kompenzácie pohybu je možné využiť kompresný pomer až 100:1 [7]. Porovnanie rozdielových snímok metódy s použitím kompenzácie pohybu a metódy bez kompenzácie je možné vidieť na obrázku č. 4.2. Neutrálne šedá farba na tomto obrázku reprezentuje nulovú mieru zmeny, ktorá následne po kompresii vhodným algoritmom nemusí byť prenášaná, čím nezaťažuje prenosové pásmo.

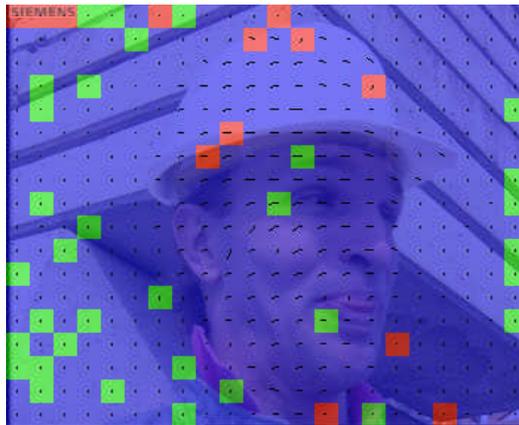


Obrázok 4.2: Znázornenie vplyvu použitia kompenzácie pohybu na obsah rozdielového snímku.

Zdroj: [http://compression.ru/video/motion\\_estimation/index\\_en.html](http://compression.ru/video/motion_estimation/index_en.html)

Princíp fungovania metód kompenzácie pohybu je založený na podobnosti jednotlivých po sebe nasledujúcich snímok. Väčšinu času sú objekty prítomné v snímanej scéne nemenné, mení sa len ich poloha v rámci daného snímky. Metóda porovnáva postupne jednotlivé oblasti dvoch alebo viacerých snímok, pričom sa

daná oblasť jednej snímky hľadá vo vyhľadávacej oblasti<sup>2</sup> druhej snímky. Pri náleze odpovedajúcom hľadanej oblasti v druhej snímke sa ukladá len informácia o posunutí, napríklad vo forme vektoru pohybu, prípadne aj rozdielové dáta ak nie je nález dokonale presný. Pre lepšiu predstavivosť sú na obrázku č. 4.3 znázornené vektory pohybu pre snímky z obrázku č. 4.1.



Obrázok 4.3: Vektory pohybu vykreslené v rekonštruovanom snímku.

Zdroj: [http://compression.ru/video/motion\\_estimation/index\\_en.html](http://compression.ru/video/motion_estimation/index_en.html)

Pri navrhovaní digitálnej stabilizácie obrazu je možné využiť rovnaký princíp vyhľadávania vektoru pohybu. Vo videozázname ovplyvnenom vibráciami kamery je celá scéna dislokovaná v jednotlivých snímkach o určitý počet bodov v určitom smere v závislosti na intenzite vibrácií. Použitím metódy odhadu a kompenzácie pohybu je možné prejav vibrácií kamery eliminovať. Podmienkou úspechu je nájdenie správneho globálneho vektoru pohybu, keďže lokálne vektory pohybu v jednotlivých oblastiach snímky sa môžu od seba značne líšiť ako je vidieť na obr. 4.3.

---

2 Okolie definovaného počtu obrazových bodov, ktoré obklopujú oblasť z prvého snímku určenú jej stredom so súradnicami  $x$  a  $y$

## 5 Návrh metódy pre stabilizáciu obrazu

Pri návrhu stabilizačnej metódy je potrebné brať do úvahy skutočnosť, že na vstupe metódy je videosekvencia v podobe série snímok a dáta o pohybe kamery nie sú dostupné. Z toho dôvodu je zvolená digitálna stabilizácia obrazu, ktorá dokáže kompenzovať vibrácie obrazu bez dodatočných informácií z pohybových senzorov.

Pre stabilizačnú metódu je možné použiť v podstate akúkoľvek zo známych metód vyhľadávania vektorov pohybu, ktorá je používaná pri kódovaní videa. Dokonca by bolo vhodné využiť prítomnosť tohoto procesu pri spracovaní videa a zakomponovať stabilizáciu obrazu priamo do neho, čím by sa odstránila nutnosť opätovného vykonávania výpočtovo náročnej operácie [8]. Keďže sa práca zameriava na využitie stabilizácie obrazu v postprodukcii, táto metóda nebude ďalej špecifikovaná.

V priebehu vytvárania práce bola na základe čiastkových výsledkov testovania a následných konzultácií s vedúcim diplomovej práce navrhnutá a implementovaná alternatívna metóda stabilizácie obrazu založená na detekcii hranice objektu. V konečnom dôsledku vykazovala v špecifických prípadoch lepšie výsledky ako metódy založené na blokovom vyhľadávaní vektorov pohybu. V nasledujúcich častiach sú postupne predstavené a popísané obe metódy.

### 5.1 Metóda stabilizácie založená na blokovom vyhľadávaní

Hlavným stavebným kameňom pôvodne navrhovanej stabilizačnej metódy bol odhad globálneho vektora pohybu a jeho využitie pri kompenzácii vibrácií obrazu.

Kompenzácia pohybu pozostáva z niekoľkých krokov:

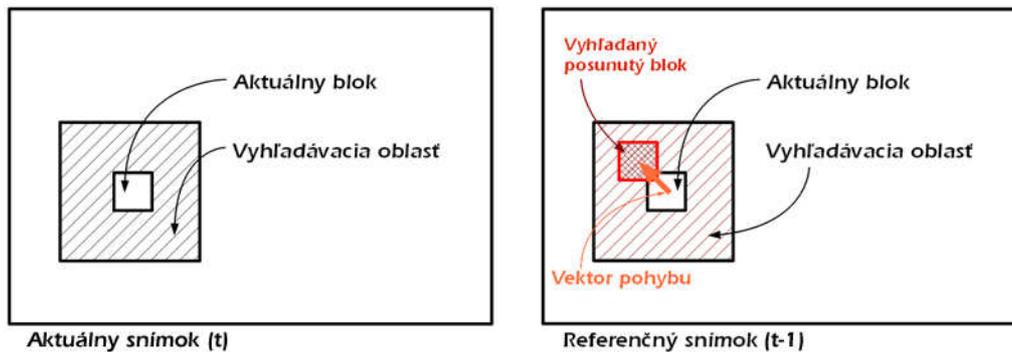
- vyhľadanie lokálnych vektorov pohybu  $LMV^3$  (viz obr. 5.1)
- výpočet globálneho vektora pohybu  $GMV^4$  z nájdených  $LMV$

---

3  $LMV$  – z anglického Local Motion Vector

4  $GMV$  – z anglického Global Motion Vector

- kompenzácia posunu snímky v smere *GMV*



Obrázok 5.1: Princíp vyhľadávania vektora pohybu.

Zdroj: Autor

Lokálne vektory pohybu sú vyhľadávané metódou *Full-Search*, ktorá všeobecne vykazuje dobrú kvalitu vyhľadávania, no vcelku vysokú výpočtovú náročnosť. Spracovávaná snímka je rozdelená pomocou pravouhlej mriežky na jednotlivé bloky. Každý blok je následne vyhľadávaný na všetkých možných pozíciách v odpovedajúcej vyhľadávacej oblasti daného bloku v referenčnom snímku (viz obr. 5.1).

Výpočet globálneho vektora pohybu je implementovaný pomocou mediánu hodnôt lokálnych vektorov. Tým je potlačený vplyv potenciálne chybné určených *LMV*, čo môže byť spôsobené prítomnosťou zvýšeného šumu v obraze.



Obrázok 5.2: Ukážka kompenzácie pohybu;

*vľavo*: snímok 1, *vpravo*: snímok 2 s kompenzáciou pohybu (snímok 2 je horizontálne prevrátený pre lepšiu viditeľnosť posuvu); okraj bez obrazových dát je zvýraznený zelenou farbou.

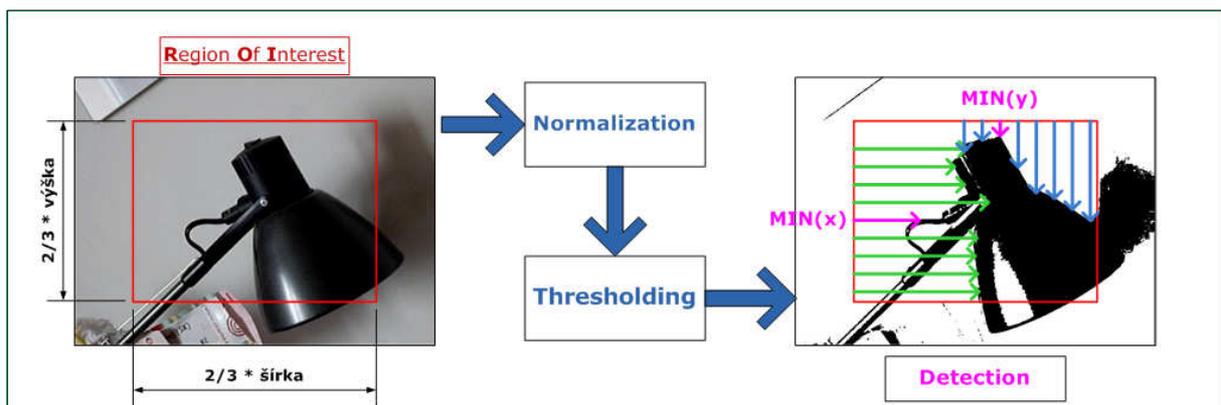
Zdroj: Autor

Následne je použitý vypočítaný *GMV* pre kompenzáciu detekovaného pohybu medzi snímkami. Na obrázku 5.2 je ukážka kompenzácie pohybu na dvoch po sebe nasledujúcich snímkach testovacej videosekvencie.

Postupne sú týmto spôsobom spracované všetky snímky privedené na vstup. Výsledkom je sekvencia stabilizovaných snímok, ktoré obsahujú na okrajoch oblasti bez platných obrazových dát (čierne pruhy). Tento efekt môže byť v určitých prípadoch nežiadúci, no je možné ho odstrániť buď čiastočným orezaním výsledných snímok, alebo extrapoláciou obrazových dát z predchádzajúcich snímok.

### 5.2 Metóda stabilizácie založená na detekcii hranice objektu

Ako už bolo uvedené v tejto práci, testovanie ukázalo, že pôvodná navrhovaná metóda založená na princípe blokového prehľadávania v niektorých prípadoch nefunguje podľa očakávaní. Z toho dôvodu bola implementovaná aj metóda stabilizácie, ktorá využíva alternatívny prístup k riešenému problému. Návrh bol založený na poznatku, že vo video-sekvenciách je väčšinou objekt záujmu udržiavaný v strede obrazového poľa, a teda je možné použitím vhodnej metódy detektovať jeho hranicu voči pozadiu scény. Obrázok 5.3 znázorňuje jednotlivé fázy procesu hľadania hranice objektu.



Obrázok 5.3: Princíp detekcie hranice objektu

Zdroj: Autor

Aby bolo možné objekt opticky oddeliť od pozadia, je na pôvodnú snímku najprv aplikovaná normalizácia jasu (*Normalization*) a následne prahovanie (*Thresholding*), vďaka čomu je ďalej spracovávaná binárna čiernobiela kópia pôvodného snímku. Na základe predpokladu, že snímaný objekt sa nachádza väčšinou v strede obrazového poľa, je v každej snímke prednastavená oblasť záujmu (*ROI*<sup>5</sup>) o veľkosti, ktorej šírka aj výška odpovedá 2/3 pôvodného rozmeru snímky so stredom v pôvodnom stredovom bode danej snímky. Postupným spracovaním jednotlivých snímok je pre každú snímku nájdený ofset objektu od okraja oblasti *ROI*. Ofset sa skladá z dvoch častí,  $ofs^x$  pre horizontálny smer a  $ofs^y$  pre smer vertikálny, ktoré sú získané jednoduchým výpočtom:

$$\begin{aligned} ofs^x &= \min(X) \\ ofs^y &= \min(Y) \end{aligned}, \text{ kde: } \begin{aligned} X &= \{x_1, x_2, \dots, x_m\}, m = \text{výška ROI} \\ Y &= \{y_1, y_2, \dots, y_n\}, n = \text{šírka ROI} \end{aligned} \quad (5.1)$$

$X$  je pole vzdialeností k detekovanej hranici na riadkoch 1, 2, ...,  $m$  (na obr. 5.3 znázornené zelenými šípkami),

$Y$  je pole vzdialeností k detekovanej hranici v stĺpcoch 1, 2, ...,  $n$  (na obr. 5.3 znázornené modrými šípkami).

Zo zistených ofsetov je určený vektor pohybu:

$$\vec{GMV} = [ ofs_t^x - ofs_{t-1}^x, ofs_t^y - ofs_{t-1}^y ] \quad (5.2)$$

pričom index  $t$  odpovedá aktuálnej snímke a index  $t-1$  snímke predchádzajúcej.

Vypočítaný vektor pohybu je následne použitý pre kompenzáciu pohybu, a to tým spôsobom, že aktuálne spracovávaný snímok je posunutý v smere určenom opačným vektorom.

Rovnako, ako v prípade použitia predchádzajúcej metódy sú výstupom snímky, v ktorých môžu byť prítomné na okrajoch čierne pruhy. Vznikajú v prípadoch, kedy daná oblasť stabilizovanej scény nebola pokrytá platnou obrazovou informáciou na

---

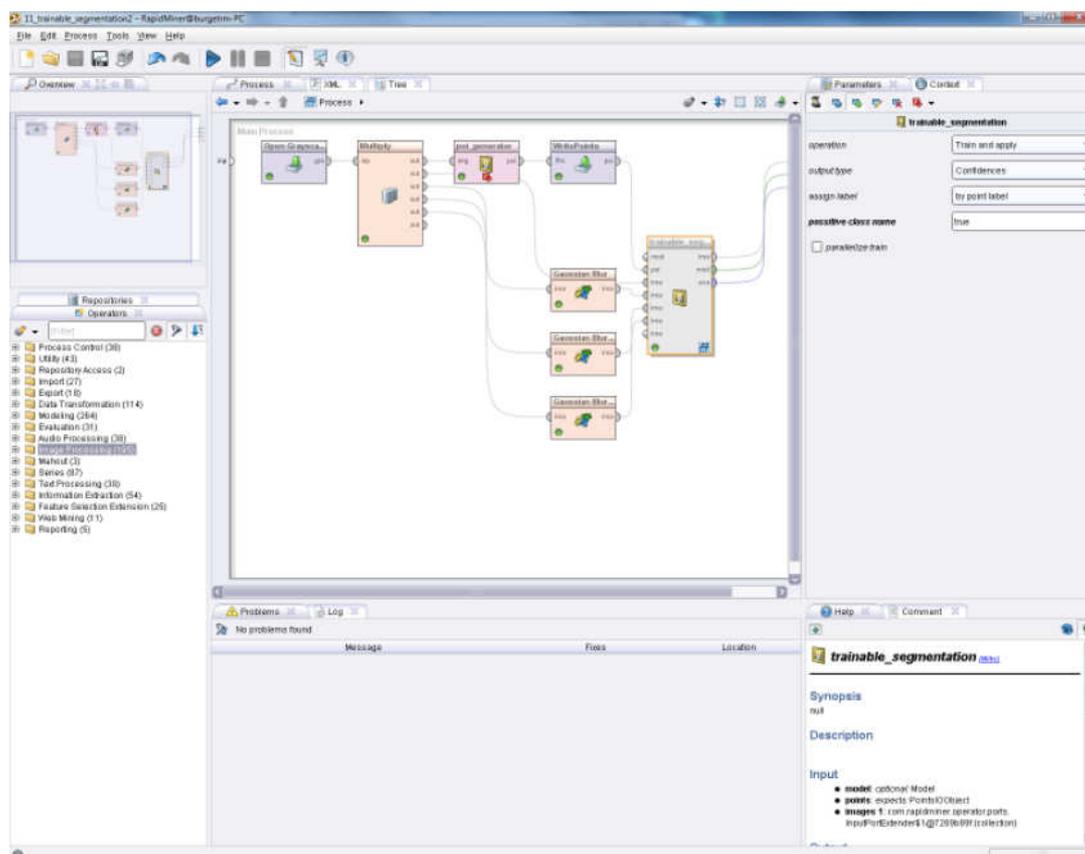
5 ROI – z anglického Region Of Interest

aktuálne spracovávanej snímke. Tento efekt môže byť v určitých prípadoch nežiadúci, no je možné ho odstrániť, ako bolo už uvedené v predchádzajúcej časti, buď čiastočným orezaním výsledných snímkov, alebo extrapoláciou obrazových dát z predchádzajúcich snímkov.

Vďaka odlišnému prístupu získavania vektoru pohybu je navrhnutá metóda výpočtovo nenáročná, pretože používa iba *dva prechody cez ROI*, jeden vo vertikálnom smere a jeden prechod v horizontálnom smere, pričom neobsahuje žiadne zložité vnorené výpočtové cykly ako je tomu pri metódach blokového vyhľadávania popísaných v predchádzajúcich kapitolách. Porovnaniu výpočtovej náročnosti rôznych vyhľadávacích metód sa podrobne venuje práca [9].

## 6 RapidMiner a rozšírenie Image Miner

RapidMiner bol vyvinutý ako nástroj pre dolovanie dát a ich analýzu. Ponúka takisto funkcie pre strojové učenie, predspracovanie dát a ich vizualizáciu, modelovanie a vyhodnocovanie. Všeobecne sa v oblasti dolovania dát pracuje so štrukturalizovanými dátami, najčastejšie spracovanými v tabuľkách, kedy každý stĺpec nesie určitý význam. Práve význam dát je kľúčovým prvkom, na základe ktorého je možné s dátami ďalej pracovať a transformovať ich na informácie s určitou hodnotou. RapidMiner je vyvíjaný ako bezplatná, flexibilná open-source platforma implementovaná v jazyku JAVA. Vďaka tomu môže fungovať na všetkých významných používaných platformách a operačných systémoch. Samotný program obsahuje grafické užívateľské prostredie (viz obr. 6.1), ktoré umožňuje prácu



Obrázok 6.1: Ukážka programového prostredia RapidMiner s návrhom procesu segmentácie obrazu.

Zdroj: <http://splab.cz/en/research/data-mining/articles>

s dátami pomocou samostatných operátorov zreťazených za sebou. Postupným spájaním jednotlivých vstupov a výstupov je možné vytvoriť komplexný proces spracovania vstupných dát. Aby bolo možné aplikovať tento princíp spracovania pre obrazové dáta, výskumná skupina SPLab<sup>6</sup> vytvorila rozšírenie do programu RapidMiner s názvom Image Mining (IMMI). Rozšírenie IMMI je prostriedkom pre prevod obrazových dát do tabuľkovej formy, s ktorou dokáže RapidMiner natívne ďalej pracovať.

Nasledujúce časti práce sú venované popisu IMMI a postupu implementácie navrhovaného operátora pre toto prostredie.

### 6.1 Predstavenie rozšírenia Image Miner (IMMI)

IMMI je rozšírením programu RapidMiner<sup>6</sup> pre získavanie znalostí z obrazových dát a ich predspracovanie.

Rozšírenie ponúka rôzne metódy pre transformáciu obrazových dát do tabuľkových dát. Tento proces prebieha tzv. výberom charakteristických znakov. Ak sú získané vlastnosti uložené v štruktúrovanej podobe (RapidMiner využíva štruktúru *ExampleSet* pre reprezentáciu štruktúrovaných dát), môže byť použitý akýkoľvek algoritmus učenia alebo algoritmus dolovania dát a práca s dátami je rovnaká ako vo všeobecných úlohách dolovania dát. Tým sú z obrazových dát získané požadované znalosti. IMMI obsahuje mnoho rôznych operátorov podporujúcich širokú škálu algoritmov pre transformácie obrazu a extrakciu príznakov [10].

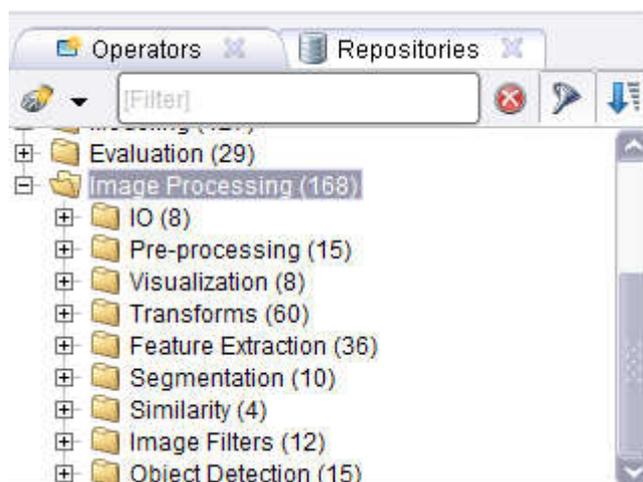
Rozšírenie obsahuje v súčasne aktuálnej verzii<sup>7</sup> viac ako 160 rôznych operátorov na prácu s obrazovými dátami, pričom sa neustále pracuje na implementácii nových

---

6 SPLab - Signal Processing Laboratory, skupina pôsobiaca na Ústave telekomunikácií, Fakulty elektrotechniky a komunikačných technológií, Vysokého učení technického v Brně; viz <http://splab.cz/>

7 V čase písania práce bola aktuálna IMMI verzia 1.4, vydaná v máji 2012 (dostupná na stiahnutie zo stránok <http://splab.cz/download/software> )

funkcií. Prehľad hlavných kategórií s uvedeným počtom prislúchajúcich operátorov je možné vidieť na obrázku 6.2.



Obrázok 6.2: Prehľad hlavných kategórií operátorov rozšírenia IMMI; čísla v zátvorkách uvádzajú počet operátorov v danej kategórii.

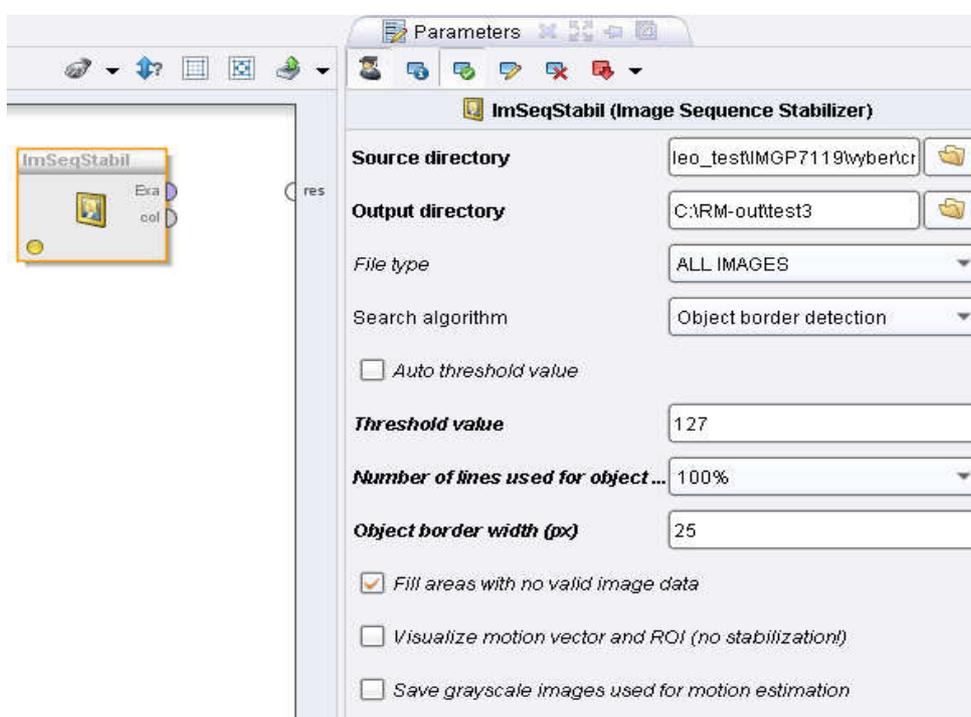
Zdroj: Autor

Viacere publikácie skupiny SPLab ukazujú možnosti praktického využitia programu RapidMiner a rozšírenia IMMI [11], [12]. Ďalšie informácie o projekte IMMI vrátane praktických návodov a ukážkových súborov je možné nájsť na webových stránkach<sup>8</sup> výskumnej skupiny SPLab.

<sup>8</sup> Webové stránky skupiny SPLab zameranej na dolovanie dát: <http://splab.cz/research/data-mining>

## 7 Praktická realizácia operátora stabilizácie obrazu

Pri návrhu operátora bol základom pôvodný koncept JAVA aplikácie. Nakoľko ale prostredie RapidMiner vyžaduje štandardizované rozhranie jednotlivých funkčných prvkov – operátorov, bolo zachované len funkčné jadro celej aplikácie, ktoré bolo nutné prispôbiť potrebám RapidMineru. Na základe konzultácií s jedným z tvorcov rozšírenia IMMI bol odsúhlasený funkčný návrh procesu stabilizácie vo forme jediného samostatného operátora s názvom *Image Sequence Stabilizer*. Ukážka užívateľského rozhrania operátora je zobrazená na obr. 7.1.



Obrázok 7.1: Ukážka užívateľského rozhrania operátora *Image Sequence Stabilizer* a jeho parametrov v prostredí RapidMiner.

Zdroj: Autor

Operátor je navrhnutý primárne pre použitie za účelom predspracovania snímok získaných konverziou videosekvencie. V mnohých video-sekvenciách je badateľné chvenie obrazu spôsobené nedostatočným fixovaním kamery počas získavania záznamu snímanej scény, ktoré je pri ďalšom spracovaní dát nežiadúce.

Na základe prevedených testov bolo overené, že pri vhodnom nastavení vstupných parametrov dokáže navrhnutý operátor chvenie potlačiť, alebo úplne odstrániť.

V nasledujúcich častiach tejto kapitoly sú popísané implementačné detaily jednotlivých metód a riešení čiastkových problémov. Celkový koncept funkčnosti popisuje časť 7.1. V časti 7.2 sú popísané jednotlivé vstupné a výstupné parametre navrhnutého operátora. Normalizácia a prevod z modelu RGB<sup>9</sup> do stupňov šedi sú vysvetlené v časti 7.3, časť 7.4 je venovaná práci s *ROI*. Časť 7.5 je venovaná jednotlivým implementovaným metódam vyhľadávania vektora pohybu *GMV*. A v posledných troch častiach tejto kapitoly sú vysvetlené podrobnosti doplnkových funkcií operátora.

### 7.1 Princíp funkčnosti operátora

Hlavný koncept procesu stabilizácie realizovanej vytvoreným operátorom je možné popísať nasledovným pseudokódom:

1. Spracovanie parametrov operátora, inicializácia pracovných adresárov, overenie vstupných súborov

Ďalej v cykle pre všetky vstupné snímky:

2. Ak nie je prvá snímka, uloženie referencie na predchádzajúcu snímku
3. Načítanie aktuálnej snímky do *ImagePlus*
4. Vytvorenie pracovnej kópie snímky v stupňoch šedi
5. Normalizácia pracovnej kópie
6. Výpočet *ROI*
7. Ak je typ algoritmu *Full-Search* alebo *Three-step Search*,
  - a) skok na krok **8.** (*motion-estimation*)
  - b) inak, skok na krok **11.** (*boundary detection*)
8. Vyhľadať pole *LMV* pomocou *motion-estimation*
9. Vyhľadať *GMV* ako medián z poľa hodnôt *LMV*
10. Skok na krok **13.**

---

9 označenie farebného modelu reprezentácie farebných snímok, ktorý pozostáva z troch farebných zložiek pre červenú, zelenú a modrú farbu, anglicky Red, Green, Blue, z čoho vznikla aj skratka RGB

11. Aplikovať prahovanie na pracovnú kópiu
12. Vyhľadať *GMV* pomocou *boundary detection*
13. Kompenzácia pohybu v smere opačnom k *GMV*
14. Vyplnenie okrajov bez platných obrazových dát
15. Výpočet *SAE*, uloženie hodnôt do *ExampleSet*
16. Uloženie upravenej snímky do výstupného adresára
17. Ak nie je posledná snímka, skok na krok **2.**
18. Zobraziť súbor výsledkov *ExampleSet*
19. KONIEC

Bližší popis metód *motion-estimation* a *boundary detection* je uvedený v časti 7.5.

### 7.2 Popis parametrov operátora *Image Sequence Stabilizer*

Vstupné parametre operátora:

- *Source directory*: cesta k adresáru so vstupnými snímkami
- *Output directory*: cesta k adresáru, do ktorého budú uložené výstupné súbory
- *File type*: typ obrazových súborov, ktoré majú byť pri stabilizácii spracované
- *Search algorithm*: výber algoritmu stabilizácie obrazu z troch dostupných implementovaných algoritmov: Full-Search, Three-step search, Object boundary detection

V prípade výberu *Object boundary detection* sú navyše dostupné tieto špecifické parametre:

- *Auto threshold value*: určuje, či bude hodnota prahu vypočítaná automaticky
- *Threshold value*: umožňuje manuálne zadať hodnotu prahu, ak nie je zvolený predchádzajúci parameter, a tým zvýšiť presnosť procesu stabilizácie
- *Number of lines used for object detection*: percentuálny podiel počtu riadkov oblasti *ROI*, ktoré sú prehľadávané pri detekcii hranice objektu

- *Object border width (px):*  
počet po sebe nasledujúcich pixelov s rovnakou jasovou hodnotou, ktoré určujú hranicu detekovaného objektu
- *Fill areas with no valid image data:*  
ponúka možnosť zapnúť alebo vypnúť vyplňanie oblastí bez platnej obrazovej informácie
- *Only visualize vectors and ROI:*  
ponúka možnosť vizualizácie oblasti *ROI* a vypočítaného vektora pohybu pre danú snímku; v prípade, že je tento parameter aktivovaný, nie je programom prevádzaná samotná kompenzácia pohybu
- *Save grayscale images used for motion estimation:*  
umožňuje zvoliť, či majú byť spolu s výstupnými farebnými snímkami ukladané aj upravené čiernobiele „pracovné“ snímky, s ktorými pracuje jadro vyhľadávacieho algoritmu (na základe ich revízie je možné optimálne upraviť hodnotu ostatných kľúčových parametrov)

Výstupný(/-é) port(-y) operátora:

- Obsah výstupnej štruktúry `ExampleSet`  
Prostredníctvom tohoto portu je na výstup posiadaná štruktúra `ExampleSet`, ktorá obsahuje výstupné dáta použitej stabilizačnej metódy.

Tabuľka 7.1: Popis polí dátovej štruktúry `ExampleSet`

	index snímku	názov vstupného súboru	názov výstupného súboru	vektor pohybu*	hodnota SAE pred kompenzáciou*	hodnota SAE po kompenzácii*
<b>dátový typ</b>	int	String	String	int[2]	double	double

\* hodnota vypočítaná voči predchádzajúcemu snímku

### 7.3 Prevod do stupňov šedi a normalizácia

Pri spracovaní farebných snímok v procese stabilizácie obrazu samotné farebné kanály nehrajú dôležitú úlohu. Ľudské oko je mnohokrát citlivejšie na vnímanie zmien jasů ako na zmeny jednotlivých farebných zložiek obrazu [13]. Navyše v tomto prípade nie je potrebné pracovať s farebnými kanálmi. Preto je vhodné po načítaní zdrojových snímok sekvencie, previesť ich farebný rozsah do odtieňov šedej. Pri prevode bol základom známy vzťah (7.1),

$$Y = 0,299 R + 0,587 G + 0,114 B \quad (7.1)$$

kde  $Y$  je výsledná hodnota jasů v šedotónovom obraze a  $R, G$  a  $B$  určujú úrovne jasů samostatných farebných zložiek. Jednotlivé koeficienty boli určené na základe rozdielnej citlivosti ľudského oka na rôzne farebné zložky svetla [14].

Normalizáciu snímku je potrebné previesť pred samotným procesom vyhľadávania vektora pohybu, aby bola znížená možnosť výskytu chyby spôsobenej rôznymi hodnotami jasovej zložky v súsledných snímkach. Navyše v prípade, ak sú na vstup privedené snímky s nízkou úrovňou kontrastu, môže normalizácia výrazne zvýšiť mieru správneho určenia vektora pohybu.

V rámci navrhnutého operátora bola normalizácia implementovaná prostredníctvom metódy rozťahnutia kontrastu. Predpokladá sa, že minimálna hodnota jasů vyskytujúca sa v snímke je  $f_1$ , maximálna hodnota  $f_2$ . Hodnoty z intervalu  $\langle f_1, f_2 \rangle$  je potrebné previesť na celý zobraziteľný rozsah hodnôt  $f \in \langle 0, 1 \rangle$ . Transformácia rozťahnutia kontrastu sa realizuje tak, že od danej hodnoty jasů  $f$  bude odčítaná minimálna hodnota nájdená v snímke a tá bude vynásobená koeficientom rozťahnutia  $1/(f_2 - f_1)$ .

Výsledkom je vzťah:

$$g = \frac{f - f_1}{f_2 - f_1} \quad (7.2)$$

kde  $g$  je upravená hodnota jasu [14]. Z programátorského pohľadu je nutné aplikovať túto transformáciu na všetky obrazové body daného snímku, aby sa dosiahol normalizovaný snímok. V knižnici ImageJ je zahrnuté rozšírenie `ContrastEnhancer`, v ktorom je už implementovaná metóda `equalize()`, ktorá prevádza túto operáciu.

#### 7.4 Určenie rozsahu a práca s oblasťou ROI

Na základe predpokladu, že sa objekt záujmu nachádza vo videosekvencii väčšinou v strede záberu a zároveň aby bola sčasti znížená výpočtová náročnosť pri použití blokového vyhľadávania, bola stanovená veľkosť ROI ako  $2/3$  pôvodnej plochy snímky (viz obr. 5.3). Na základe toho je nutné pre každú snímku vypočítať počiatkové súradnice okna ROI, jeho šírku a výšku. S ohľadom na zámer použitia blokových vyhľadávacích algoritmov, sú navyše upravené tieto rozmery tak, aby šírka aj výška bola najbližším nižším celočíselným násobkom veľkosti použitého bloku<sup>10</sup>.

Aby bolo možné jednoducho pracovať s ROI v danej snímke, bola použitá na extrakciu odpovedajúcich obrazových bodov trieda `Jama.Matrix` z knižnice JAMA<sup>11</sup>. Táto trieda umožňuje prácu s dátami v maticovej reprezentácii, pri použití ktorého dochádza k zjednodušeniu operácií pri spracovávaní obrazových 2-D dát.

#### 7.5 Implementované metódy vyhľadávania vektoru pohybu

Spočiatku bola navrhovaná metóda stabilizácie obrazu založená na princípe blokového vyhľadávania zhody, tak ako algoritmy určené na kódovanie videa. Pri tomto procese boli implementované dva rôzne vyhľadávacie algoritmy – *Full-Search* a *Three-step search*, ktorých bližší popis je uvedený v samostatných častiach ďalej v tejto práci. Po implementácii týchto dvoch algoritmov bolo pri testovaní zistené, že

---

<sup>10</sup> Veľkosť bloku je určená konštantou `BLOCK_SIZE` a je prednastavená na 16 pixelov.

<sup>11</sup> JAMA : A Java Matrix Package; dostupné z: <http://math.nist.gov/javanumerics/jama/>

v niektorých prípadoch nefungujú podľa očakávania. Preto bola navrhnutá a implementovaná alternatívna metóda vyhľadávania vektoru pohybu, popísaná ďalej v časti 7.5.3, základom ktorej je detekcia hranice objektu v oblasti *ROI*.

### 7.5.1 *Full-Search*

Pre potreby práce s obrazovými blokmi bola v tejto metóde pre uchovanie obrazových dát použitá knižnica JAMA<sup>12</sup>. Trieda `Jama.Matrix` ponúka rozhranie a metódy pre prácu s maticovým usporiadaním dát, čo je dôležité pre efektívnu prácu s jednotlivými blokmi obrazu. Na vstup tejto metódy sú privedené dve šedotónové snímky, aktuálna a predchádzajúca (referenčná). *Full-Search* prechádza oblasťou *ROI* aktuálnej snímky postupne po makroblochoch s veľkosťou 16x16 pixelov. Pre každý makroblok sa snaží vyhľadať najpodobnejší blok v referenčnej snímke. Vyhľadávanie prebieha v rámci vyhľadávacej oblasti, ktorej šírka a výška odpovedajú dvojnásobku šírky makrobloku, tzn. oblasť s rozmermi 32x32 pixelov. Z uvedených rozmerov zároveň vyplýva maximálny rozsah  $\langle -8, +8 \rangle$  pixelov, v rámci ktorého je možné použitím tejto metódy kompenzovať pohyb medzi nasledujúcimi snímkami.

Pre každý makroblok je prevádzaný nasledovný postup:

1. Určiť vyhľadávaciu oblasť v referenčnej snímke pre daný makroblok.
2. Pre všetky možné polohy referenčného makrobloku vypočítať mieru chyby (hodnota SAE).
3. Pre referenčný makroblok s najnižšou hodnotou SAE je zaznamenaná pozícia, z ktorej je určený lokálny vektor pohybu LMV.
4. Zaznamenať LMV do poľa hodnôt.
5. Ak nie je posledná snímka,
  - a) skoč na krok **2**.
  - b) inak pokračuj
6. Nájsť medián z uložených hodnôt LMV a odoslať ju na výstup ako GMV.

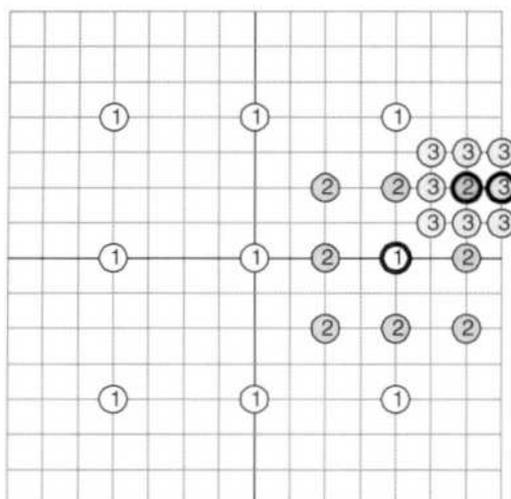
Výsledný vektor pohybu GMV je použitý pre kompenzáciu vychýlenia scény.

Nakoľko je vyhľadávací algoritmus *Full-search* veľmi výpočtovo náročný, bol pre porovnanie implementovaný aj rýchlejší a menej náročný *Three-step search*.

### 7.5.2 *Three-step search*

Podobne ako pri algoritme *Full-Search*, bola použitá knižnica JAMA<sup>12</sup> pre efektívnu prácu s maticovými dátami. Rovnaké sú aj vstupné parametre tejto metódy.

Hlavný princíp vyhľadávania je rovnaký ako pri algoritme *Full-Search*, no rozdiel je v rozsahu prehľadávania vyhľadávacej oblasti referenčnej snímky. Nie sú prehľadávané všetky možné pozície, ale len pevne daný počet postupne v troch



Obrázok 7.2: Princíp prehľadávania algoritmu *Three-step search*.

Zdroj: <http://blog.weisu.org/2008/12/search-methods-in-motion-estimation.html>

krokoch. V každom kroku sa postupne znižuje ofset pre pozície referenčného makrobloku. Spôsob vyhľadávania je lepšie pochopiteľný pri pohľade na obrázok 7.2. Každý krok počíta hodnotu *SAE* v bodoch tzv. osemokolia stredového bodu daného makrobloku.

Bod s najnižšou hodnotou *SAE* je vybraný ako nový stred vyhľadávania pre nasledujúci krok. V záverečnom kroku sú prehľadávané makrobloky už len v bezprostrednej blízkosti stredového bodu. Tento prístup zaručuje, že bude pre každý makroblok prehľadaných práve 25 možných pozícií v referenčnej snímke.

Následné spracovanie je zhodné so spracovaním po použití algoritmu Full-Search – uložené lokálne vektory  $LMV$ , nájdený medián z týchto hodnôt a aplikovaný pri kompenzácii pohybu ako  $GMV$ .

### 7.5.3 Object border detection

Určenie vektora pohybu v tejto metóde závisí na vyhľadani hranice objektu vnútri  $ROI$ . Predpokladom správnej funkčnosti je hlavná podmienka, aby objekt, ktorý má byť detekovaný nepresahoval hranice  $ROI$ . Vektor pohybu je určený na základe zmeny polohy objektu v rámci  $ROI$ . Na vstup tejto metódy sú privedené dve binárne<sup>12</sup> snímky, ktoré sú výstupom operácií normalizácie a prahovania aplikovaných na šedotónové snímky získané z  $RGB$ . Vďaka binárnej forme je možné rozpoznať hranicu objektu jednoduchým prechádzaním poľa hodnôt reprezentujúcich určitý riadok respektíve stĺpec snímky, pričom je kontrolovaný prechod z jednej hodnoty na druhú, tzn.  $0 \rightarrow 1$ , respektíve  $1 \rightarrow 0$ . Aby nedochádzalo ku chybným detekciám spôsobovaným prítomnosťou šumu v snímkach, bol zavedený vstupný parameter „hrúbka hranice“. Pomocou neho sú odfiltrované detekcie, pri ktorých hrúbka hranice nedosahuje požadované minimum.

Pre každú snímku metóda prevádza nasledovný postup:

Pre všetky riadky  $ROI$ :

1. Uložiť hodnoty pixelov riadku aktuálnej snímky do poľa hodnôt.
2. Vyhľadať hranicu objektu na danom riadku.
3. Uložiť hodnoty predchádzajúcej snímky do poľa hodnôt.
4. Vyhľadať hranicu objektu v predchádzajúcej snímke.
5. Uložiť hodnoty do zoznamu.
6. Ak nie je posledná snímka,
  - a) skok na krok **1.** (presun na ďalší riadok),
  - b) inak pokračovať.

---

<sup>12</sup> Binárna snímka – hodnoty jednotlivých bodov môžu nadobudnúť len jednu z hodnôt  $\{0, 1\}$

7. Výber minima zo zoznamu vzdialeností hraníc pre aktuálnu a predchádzajúcu snímku.
8. Rozdielom minimálnych hodnôt dostávame horizontálnu zložku vektora pohybu

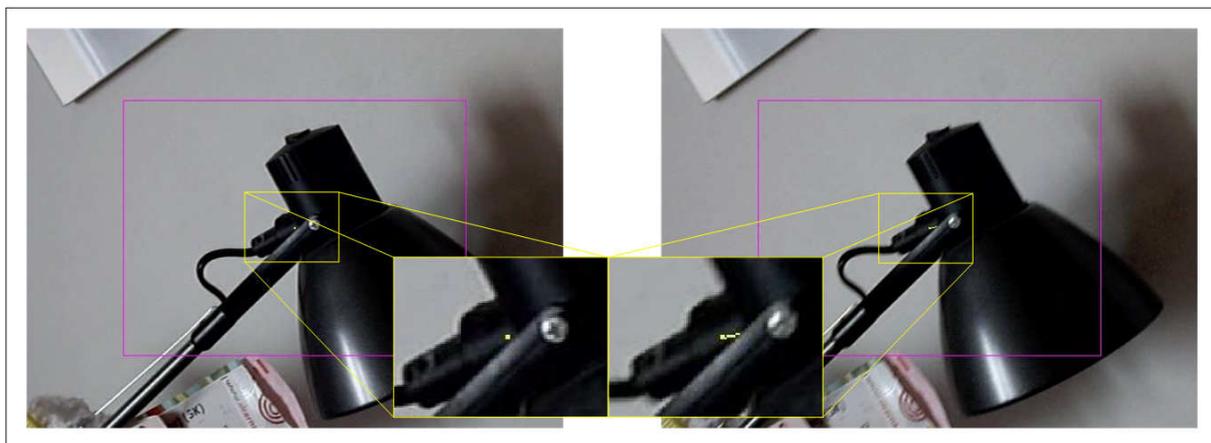
Proces opakujeme pre všetky stĺpce ROI:

*Proces je analogický k vyššie popísanému, rozdielom nájdených minimálnych hodnôt aktuálnej a predchádzajúcej snímky dostávame vertikálnu zložku vektora pohybu.*

Výstupom tejto metódy je celkový vektor pohybu, ktorý je ďalej použitý štandardným spôsobom pre kompenzáciu pohybu.

### 7.6 Vykresľovanie vektoru pohybu a oblasti ROI

Pre testovacie účely bola do operátora implementovaná aj funkcia, ktorá znázorňuje reálny rozsah ROI a vypočítaný vektor pohybu pre daný snímok. Táto funkcia sa aktivuje pomocou parametra „*Only visualize vectors and ROI*“. Aktivácia tejto funkcionality zároveň zakáže reálnu stabilizáciu obrazu, aby bola na výstupných snímkoch lepšie viditeľná miera presnosti určenia vektora pohybu. Ukážka takéhoto výstupu je na nasledujúcom obrázku 7.3.



Obrázok 7.3: Ukážka vykresľovania vektoru pohybu a oblasti ROI.

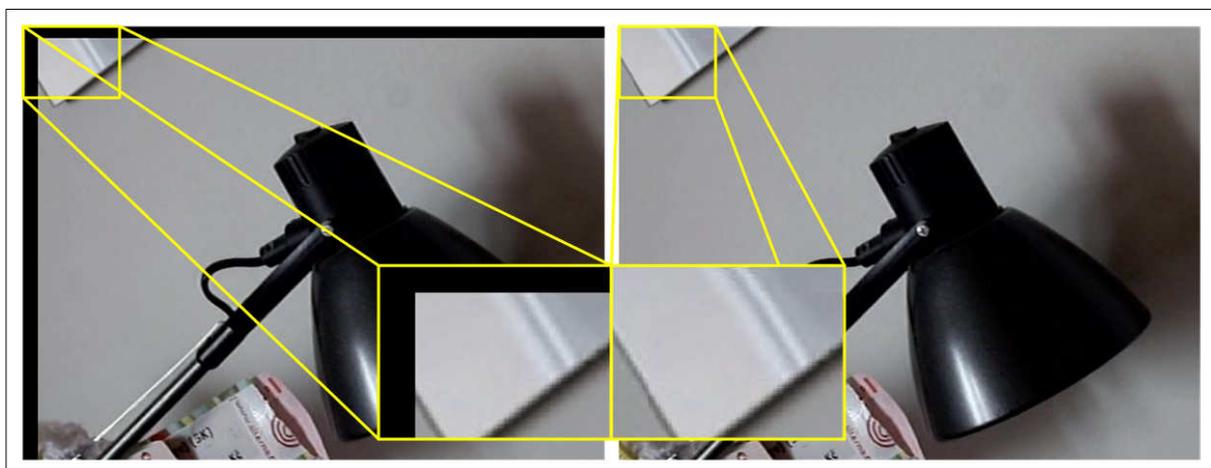
Zdroj: Autor

Samotné vykresľovanie je riešené vstavanými metódami triedy `ImageProcessor` z knižnice `ImageJ`. Oblasť ROI je vykreslená farbou **purpurovou**, stred snímku s vektorom pohybu sú vykreslené farbou **žltou**.

### 7.7 Vypĺňanie oblastí bez platnej obrazovej informácie

Implementácia riešenia je založená na princípe ukladania tzv. kľúčových snímok, ktoré sú vykresľované na pozadí. Do popredia je potom vykreslená aktuálna vychýlená snímka. Kľúčovou snímkou je každá 10. snímka v poradí. Za predpokladu korektnej funkčnosti metódy vyhľadávania vektora pohybu a faktu, že je snímaná scéna statická<sup>13</sup>, nie sú vo výstupných snímkach badateľné oblasti bez obrazovej informácie a rekonštruované oblasti plynule nadväzujú na posunuté obrazové dáta.

Porovnanie výstupu stabilizácie s vypnutou a zapnutou funkciou vypĺňania je možné vidieť na obrázku 7.4.



Obrázok 7.4: Ukážka vypĺňania okrajových oblastí snímky bez platných obrazových dát  
Zdroj: Autor

### 7.8 Výpočet hodnoty MAE pre upravené snímky

S cieľom vytvorenia ukazovateľa, ktorý by reflektoval mieru správnej funkčnosti navrhnutej metódy je pre každú snímku vypočítaná hodnota  $MAE^{14}$  porovnaním s predchádzajúcou snímkou podľa vzťahu:

$$MAE = \frac{1}{m * n} \sum_{i=1}^m \sum_{j=1}^n |act_{ij} - prev_{ij}| , \quad (7.3)$$

<sup>13</sup> Operátor je určený pre stabilizáciu statických scén.

<sup>14</sup> MAE – z anglického Mean Absolute Error

kde  $m$  je šírka a  $n$  je výška snímky, prípadne jej časti, ktorá je predmetom záujmu,  $act_{ij}$  je hodnota jasú aktuálnej snímky v bode so súradnicami  $i, j$  a  $prev_{ij}$  je hodnota jasú predchádzajúcej snímky na rovnakých súradniciach [15]. Pre každú snímku sú zaznamenané do výsledkov dve vypočítané hodnoty  $MAE$ . Prvá odpovedá stavu pred kompenzáciou pohybu, druhá je vypočítaná až po jej prevedení. Kontrolou týchto hodnôt je možné následne určiť do akej miery bola prevedená stabilizácia presná, prípadne je pomocou nich možné porovnať jednotlivé stabilizačné metódy aplikované na rovnakú videosekvenciu a určiť, ktorá je pre daný prípad najvhodnejšia.

## 8 Testovanie a výsledky

Operátor stabilizácie obrazu bol vytváraný v jazyku JAVA s použitím Java Runtime Environment verzie 6 update 32 (build 1.6.0\_32-b05). Aby bolo možné vyvíjať operátory pre rozšírenie *IMMI*, je potrebné použitie vývojového prostredia s podporou SVN<sup>15</sup>. V rámci tejto práce bolo použité prostredie *Eclipse*. Pre *Eclipse* zabezpečuje SVN funkcionality rozšírenie *Subclipse*<sup>16</sup>. Samotný RapidMiner bol nainportovaný do prostredia *Eclipse* ako pracovný projekt z voľne dostupného SVN úložiska<sup>17</sup> prevádzkovaného firmou Rapid-I<sup>18</sup>. Návod na import projektu je dostupný buď na stránkach skupiny SPLab6, alebo priamo na stránkach výrobcu softwaru – firmy Rapid-I.

Testy boli prevádzané na systéme HP ProBook s nasledovnými parametrami:

- Procesor: Intel(R) Core(TM) i5 M450 @ 2,4GHz
- Pamäť RAM: 3,80 GB použiteľných
- Operačný systém: Microsoft Windows 7 Professional (64bitový)  
Service Pack 1

### 8.1 Videosekvencie

Pre účely testovania navrhnutého operátora boli vytvorené testovacie videosekvencie so štandardným VGA rozlíšením. Testovacie videosekvencie vznikli na zariadení Pentax Optio A10, ktoré patrí do kategórie bežných užívateľských kompaktných fotoaparátov s podporou pre nahrávanie videa. Aby bol v obraze zviditeľnený jav chvenia obrazu prenášaný z ruky osoby pracujúcej s prístrojom, boli testovacie videosekvencie získané pri maximálnom dostupnom optickom priblížení.

---

15 SVN – Apache *Subversion* je systém pre správu a verzovanie zdrojových kódov

16 Rozšírenie podporujúce SVN pre Eclipse; dostupné z: <http://subclipse.tigris.org/>

17 RapidMiner VEGA dostupný cez SVN z: <https://rapidminer.svn.sourceforge.net/svnroot/rapidminer/Vega>

18 Bližšie informácie o Rapid-I sú dostupné na stránkach: <http://rapid-i.com>

Zároveň bolo nutné vypnúť vstavanú funkciu optickej stabilizácie obrazu s názvom *Shake reduction system*, aby nedošlo k nechcenej stabilizácii ešte pred samotným spracovaním pomocou vytvoreného operátora. Na prevedenie videosekvencií na postupnosť snímok bol použitý program *VirtualDub*<sup>19</sup>.

Len pre zaujímavosť uvádzam, že podľa dostupných informácií sa v prípade stabilizačnej funkcie *Shake reduction system*, ktorá je vstavaná do prístroja, jedná o variantu optickej stabilizácie so vstavaným pohyblivým čipom, ktorá bola spomínaná v časti 3.2.

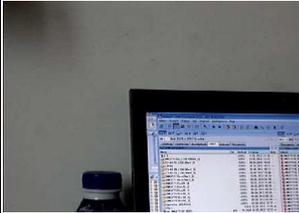
Vytvorené videosekvencie, v ktorých je viditeľné chvenie obrazu je možné nájsť v dátovej prílohe na CD nosiči. Ich prehľad je uvedený v tabuľke 8.1.

## 8.2 Dosiahnuté výsledky

Prehľad hodnôt MAE jednotlivých videosekvencií vypočítaných pri testovaní je uvedený v tabuľke 8.1. Z dosiahnutých výsledkov je zrejmé, že navrhnutý systém funguje najlepšie pre scény, kde je hlavný objekt na kontrastnom pozadí umiestnený v oblasti *ROI*. V prvej sekvencii s názvom „IMGP7114“ je miera zníženia *MAE* len 7,07%, čo je jasným ukazovateľom nesprávnej detekcie chvenia v obraze. V prípade správnej detekcie a kompenzácie chvenia obrazu sa hodnota zníženia *MAE* pohybuje v rozsahu približne 65 – 85 %. Najvýraznejšie zníženie chyby bolo dosiahnuté v testovacej sekvencii „IMGP7119“ a dosiahlo hodnoty 83,6 %. Vo filmovanej scéne sa nachádza jednofarebný tmavý objekt na svetlom pozadí. Pravdepodobne vďaka tomu bol detekovaný jeho pohyb medzi jednotlivými snímkami s najmenšou chybou práve v tejto videosekvencii.

---

19 <http://www.virtualdub.org>

Ukážka	Názov	InitMAE			FinalMAE			Zníženie MAE
		min	max	avg	min	max	avg	
	IMPGP7114	0,16	27,31	<b>11,95</b>	0,16	27,31	<b>11,10</b>	<b>7,07%</b>
	IMPGP7115	0,41	40,08	<b>24,82</b>	0,26	35,79	<b>8,19</b>	<b>66,99%</b>
	IMPGP7116	3,37	46,39	<b>30,06</b>	0,22	34,66	<b>6,42</b>	<b>78,66%</b>
	IMPGP7119	0,73	20,49	<b>11,02</b>	0,40	6,21	<b>1,81</b>	<b>83,60%</b>
	IMPGP7121	0,56	19,21	<b>14,16</b>	0,34	8,06	<b>2,39</b>	<b>83,09%</b>

Tabuľka 8.1: Prehľad miery zníženia chyby MAE v jednotlivých videosekvenciách dosiahnutej navrhnutou metódou stabilizácie obrazu.

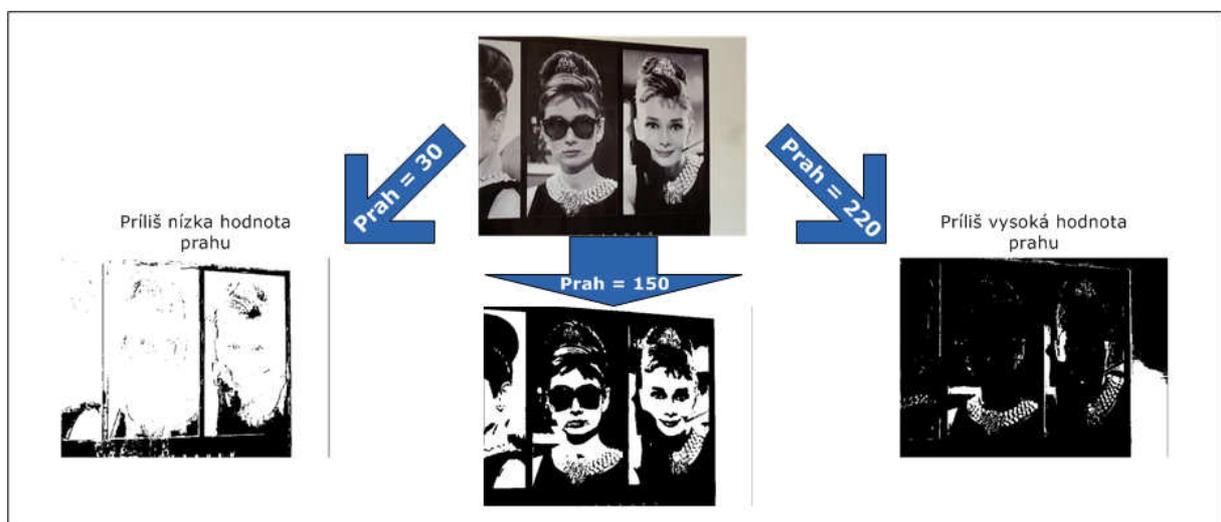
### 8.3 Nastavovanie hodnoty prahu

Parameter pre nastavovanie prahovej hodnoty, ktorý je súčasťou metódy *Object border detection*, plní kľúčovú úlohu v procese stabilizácie. Implicitne je nastavené automatické prahovanie, ktoré dobre funguje pre snímky s dobre rozlíšiteľným objektom záujmu v scéne. Dobrá rozlíšiteľnosť v prenesenom význame znamená, že

jasová zložka odpovedajúca objektu záujmu je od jasovej zložky prislúchajúcej pozadiu scény oddelená vysokým kontrastom. Na základe nastavenej hodnoty prahu sa menia vstupné dáta, s ktorými pracuje stabilizačná metóda.

Pri príliš nízkej hodnote prahu bude pracovná čiernobiela snímka vyplnená bielou farbou, prípadne sa môžu v jednotlivých snímkach vo zvýšenej miere prejavovať artefakty vznikajúce kvôli aplikovanej kompresii.

Pre nevhodne vysokú hodnotu prahu bude situácia podobná avšak opačná – prevládať bude na pracovných snímkach čierna farba, môžu sa vyskytovať biele artefakty (viz obrázok 8.1).



Obrázok 8.1: Ukážka vplyvu nastavenej hodnoty prahu na výstupný snímok

Keďže každá nafilmovaná videosekvencia môže byť iná, nie je možné určiť všeobecne platnú hodnotu prahu, ktorá by bola optimálna pre všetky prípady. Implicitne je hodnota prahu nastavená na 127, čo odpovedá strednej hodnote z rozsahu  $\langle 0, 255 \rangle$ . Pri testovaní sa však osvedčil nasledovný postup kombinujúci automatické a manuálne nastavenie parametra prahu.

- Najprv je spustené spracovanie s automatickým prahovaním.
- V prípade neuspokojivých výsledkov je následne možné z výstupu procesu zistiť vypočítanú hodnotu prahu pre danú skupinu snímok.

- Po vizuálnej kontrole uložených pracovných snímok (dostupných po zaškrtnutí príslušného parametra v možnostiach operátora) by malo byť užívateľovi jasné, v akom smere je potrebné korigovať hodnotu prahu. (viz obr. 8.1).

#### 8.4 Rozsah riadkov a stĺpcov

V prípade, že je snímaná scéna, kde objekt záujmu má pravidelný obdĺžnikový tvar, ktorého strany sú rovnobežné s okrajmi scény, je možné pomocou parametra definujúceho rozsah použitých riadkov a stĺpcov znížiť výpočtovú náročnosť a urýchliť tak výpočet.

#### 8.5 Výkonnostné testy

Pre testovacie účely boli do programu implementované základné nástroje pre meranie využitia procesorového času. V tabuľke 8.2 je možné vidieť využitie procesorového času v závislosti na použití jednotlivých výpočtových metód. Testovanie bolo prevedené na dvoch rôznych sekvenciách snímok. Parametre systému, na ktorom bolo testovanie prevádzané sú uvedené v úvode tejto kapitoly.

Použitý algoritmus	Full-search	Three-step search	Object Border Detection	Full-search	Three-step search	Object Border Detection
	Procesorový čas (s) pre výpočet			Procesorový čas (s) pre súborové operácie		
Videosekvencia	Procesorový čas (s) pre výpočet 1 snímky			Proc.čas (s) pre súborové operácie na 1 snímok		
IMGP7121	74,58	19,06	8,36	13,04	13,30	13,44
120 snímok	0,62	0,16	0,07	0,11	0,11	0,11
IMGP7116	46,63	11,44	4,94	7,67	7,82	7,66
76 snímok	0,61	0,15	0,07	0,10	0,10	0,10

Tabuľka 8.2: Znáznornenie výpočtovej náročnosti jednotlivých metód pomocou spotrebovaného procesorového času.

## 9 Záver

V tejto práci sú uvedené základné informácie o probléme stabilizácie obrazu. Boli zmapované možnosti riešenia tohoto problému, ktoré sú dostupné v dnešnej dobe. Podľa zadaných kritérií boli vytvorené testovacie videosekvencie, v ktorých je patrné chvenie obrazu. Na základe špecifických požiadaviek bol predstavený návrh vlastnej stabilizačnej metódy, ktorá bola následne implementovaná do prostredia RapidMiner. Vytvorený operátor pre prostredie RapidMiner umožňuje do istej miery eliminovať vibrácie viditeľné v obraze. V práci boli postupne predstavené dve rôzne alternatívy riešenia.

Prvá navrhnutá alternatíva pri testovaní vykazovala určité chyby, preto bola navrhnutá ďalšia metóda založená na odlišnom prístupe. Testovaním bolo overené, že navrhnuté riešenie dokáže pri správnom nastavení vstupných parametrov znížiť mieru chyby medzi snímkami o 65 – 85 %.

Výstupom spracovania je stabilizovaná sekvencia snímok a sada metadát získaných počas procesu stabilizácie, ktoré môžu byť ďalej spracovávané v prostredí RapidMiner.

Ďalším rozšírením práce by mohla byť implementácia nástroja, ktorý by automatizoval nastavovanie vstupných parametrov pre navrhnutý operátor, čo by mohlo viesť k dosiahnutiu vyššej kvality výsledkov navrhnutej metódy.

## 10 Zoznam použitej literatúry

- [1]: Ken-Lab: Gyro Kits. Kenyon Laboratories [online]. 2011 [cit. 2012-05-23]. Dostupné z: [http://www.ken-lab.com/mm5/merchant.mvc?Screen=CTGY&Store\\_Code=KL&Category\\_Code=GK](http://www.ken-lab.com/mm5/merchant.mvc?Screen=CTGY&Store_Code=KL&Category_Code=GK)
- [2]: SACHS, David, NASIRI a Daniel GOEHL. Image Stabilization Technology Overview. Image Stabilization Technology Overview [online]. 2006 [cit. 2012-05-23]. Dostupné z: [http://invensense.com/mems/gyro/documents/whitepapers/ImageStabilizationWhitepaper\\_051606.pdf](http://invensense.com/mems/gyro/documents/whitepapers/ImageStabilizationWhitepaper_051606.pdf)
- [3]: SHAKOOR, M.H. a A.R. DEHGHANI. Fast Digital Image Stabilization By Motion Vector Prediction. In: 2nd Conference on Environmental Science and Information Application Technology. Shiraz, Iran: Islamic Azad University, 2010.
- [4]: JOSHI, Neel, Sing Bing KANG, C. Lawrence ZITNICK a Richard SZELISKI. Image Deblurring using Inertial Measurement Sensors: Microsoft Research. Image Deblurring using Inertial Measurement Sensors [online]. 2010 [cit. 2012-05-23]. Dostupné z: [http://research.microsoft.com/en-us/um/redmond/groups/iom/imudeblurring/imu\\_deblurring.pdf](http://research.microsoft.com/en-us/um/redmond/groups/iom/imudeblurring/imu_deblurring.pdf)
- [5]: READ, Paul a Mark-Paul MEYER. Restoration of motion picture film. Boston: Butterworth-Heinemann, 2000, viii, 359 p. ISBN 07-506-2793-X.
- [6]: VirtualDub MSU Motion Estimation Filter. VATOLIN. MSU GRAPHICS AND MEDIA LAB. Everything about the data compression [online]. 2011, 10.3.2011 [cit. 2012-05-23]. Dostupné z: [http://compression.ru/video/motion\\_estimation/index\\_en.html](http://compression.ru/video/motion_estimation/index_en.html)
- [7]: Motion estimation. In: ZALLER, Joe. Broadcast Engineering [online]. 2007, 1.8.2007 [cit. 2012-05-23]. Dostupné z: [http://broadcastengineering.com/storage\\_networking/motion-estimation/](http://broadcastengineering.com/storage_networking/motion-estimation/)
- [8]: CHEN, H., C. LIANG, Y. PENG a H. CHANG. Integration of digital stabilizer with video codec for digital video cameras. IEEE Trans. Circuits Syst. Video Technol. 2007, vol. 17, č. 7, s. 801-813.
- [9]: PEINSIPP, Holger. Implementation of a Java applet for demonstration of block-matching motion-estimation algorithms [online]. University of Mannheim, 2003 [cit. 2012-04-24]. Dostupné z: [http://pi4.informatik.uni-mannheim.de/pi4.data/content/animations/motionestimation/Studienarbeit\\_Motion\\_Estimation.pdf](http://pi4.informatik.uni-mannheim.de/pi4.data/content/animations/motionestimation/Studienarbeit_Motion_Estimation.pdf). Studienarbeit. University of Mannheim. Vedúci práce Dirk Farin.
- [10]: BURGET, R., KARÁSEK, J., SMĚKAL, Z., UHER V., DOSTÁL, O., RapidMiner Image Processing Extension: A Platform for Collaborative Research, International Conference on TELECOMMUNICATIONS AND SIGNAL PROCESSING, Baden Austria 2010

[11]:BURGET, R.; ČÍKA, P.; ZUKAL, M.; MAŠEK, J. *Automated Localization of Temporomandibular Joint Disc in MRI Images*. In *2011 34th International Conference on Telecommunications and Signal Processing (TSP)*. 2011. p. 413-416. ISBN: 978-1-4577-1409-2

[12]:BURGET, R.; ŠMIRG, O.; KERKOVSKÝ, M.; SMÉKAL, Z.; ŠPRLÁKOVÁ, A. *Automatic segmentation of focal lesions in the brain using artificial intelligence methods*. In *13th International Conference on Research in Telecommunication Technologies*. Těchov: 2011.p. 179-182. ISBN: 978-80-214-4283-2

[13]: GONZALEZ, Rafael C. a Richard E. WOODS. *Digital image processing*. 3rd ed. Upper Saddle River, N.J: Pearson Education, 2008. ISBN 01-350-5267-X.

[14]: DOBEŠ, Michal. *Zpracování obrazu a algoritmy v C#*. 1. vyd. Praha: BEN - technická literatura, 2008, 143 s. ISBN 978-80-7300-233-6.

[15]: FURHT, Borivoje. *Encyclopedia of multimedia*. 2nd ed. New York: Springer, 2008, 1001 s. ISBN 978-038-7784-151.