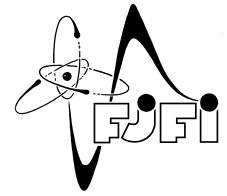


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta jaderná a fyzikálně inženýrská



Automatická registrace snímků video

Automatic registration of video frames

Bakalářská práce

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Daniela Lichnovská
Studijní program: Aplikace přírodních věd
Studijní obor: Matematické inženýrství
Studijní zaměření: Aplikované matematicko-stochastické metody
Název práce (česky): Automatická registrace snímků videa
Název práce (anglicky): Automatic registration of video frames

Pokyny pro vypracování:

- 1) Seznamte se základními postupy registrace obrazů a skládání panoramat.
- 2) Navrhněte registrační metodu vhodnou na skládání snímků videa zachycující regál v obchodě.
- 3) Navrhněte metodu na detekci kvality registrace.
- 4) Analyzujte vliv rozmazání způsobené pohybem ruky na kvalitu registrace.
- 5) Navrhněte metodu na výběr snímků z videa optimalizující prostorové pokrytí regálu a kvalitu registrace.
- 6) Experimentálně ověřte navržený postup na reálných videích.
- 7) V případě, že bude k dispozici metoda na detekci zboží, tak ověřte funkčnost navrženého postupu i s ohledem na schopnost detektovat správně zboží.

Doporučená literatura:

- 1) R. C. Gonzales, R. E. Woods, Digital Image Processing (2nd ed.). Prentice Hall, 2002.
- 2) B. Zitová, J. Flusser, Image registration methods: a survey. Image and Vision Computing 21, 2003, 977-1000.
- 3) J. Mustaniemi, J. Kannala, S. Särkkä, J. Matas, J. Heikkilä, Fast motion deblurring for feature detection and matching using inertial measurements. In 'IEEE ICPR', Beijing, China, 2018.

Jméno a pracoviště vedoucího bakalářské práce:

doc. Ing. Filip Šroubek, Ph.D. DSc.
ÚTIA AV ČR Praha, v.v.i., Pod Vodárenskou věží 4, 182 00 Praha 8

Jméno a pracoviště konzultanta:

Ing. Lukáš Tryner
DataSentics, a.s., Washingtonova 1599/17 , Praha 1

Datum zadání bakalářské práce: 31.10.2020

Datum odevzdání bakalářské práce: 7.7.2021

Doba platnosti zadání je dva roky od data zadání.

V Praze dne 30.10.2020

RB

garant oboru

R. Hlaváč

vedoucí katedry



děkan

Poděkování:

Chtěl bych zde poděkovat především svému školiteli za pečlivost, ochotu, vstřícnost a od-
borné i lidské zázemí při vedení mé diplomové práce. Dále děkuji svému konzultantovi za
.....

Čestné prohlášení:

Prohlašuji, že jsem tuto práci vypracoval samostatně a uvedl jsem všechnu použitou literaturu.

V Praze dne 16. dubna 2021

Daniela Lichnovská

Název práce:

Název práce

Autor: Daniela Lichnovská

Obor: Matematické inženýrství

Zaměření: Aplikované matematicko-stochastické metody

Druh práce: Bakalářská práce

Vedoucí práce: doc. Ing. Filip Šroubek, Ph.D. DSc., ÚTIA AV ČR Praha, v.v.i., Pod Vodárenskou věží 4, 182 00 Praha 8

Konzultant: Ing. Lukáš Tryner, DataSentics, a.s., Washingtonova 1599/17, Praha 1

Abstrakt: Abstrakt max. na 10 řádků. Abstrakt max. na 10 řádků.

Klíčová slova: klíčová slova (nebo výrazy) seřazená podle abecedy a oddělená čárkou

Title:

Automatic registration of video frames

Author: Daniela Lichnovská

Abstract: Max. 10 lines of English abstract text. Max. 10 lines of English abstract text.

Key words: keywords in alphabetical order separated by commas

Obsah

Úvod	8
1 Registrace obrazu	9
1.0.1 co to je	9
1.1 Příznaky a jejich detekce	9
1.2 Korespondence příznaků	9
1.3 Odhad transformace	10
1.3.1 RANSAC	10
1.3.2 Afinní a projektivní transformace	10
1.4 transformace(pipelina?)	10
2 Výběr snímků z videa	11
2.1 focus measure	11
2.2 Velikost okna	11
2.3 Error threshold	11
2.4 pipeline	12
2.5 Registrace panoramu	12
2.5.1 computeTforms	12
2.5.2 computeLimits	12
2.5.3 createPanorama	12
2.5.4 Omezení	12
3 Experimenty	13
3.1 Opakovatelnost(repeatability)	13
3.2 Vliv rozmazání na registraci	14
3.3 vliv artefaktů	14
3.4 Detekce cenovek	14
3.5 Omezení	16
3.6 Výsledky	16
Závěr	18

Úvod

Tuto bakalářskou práci jsem vypracovala ve spolupráci se společností DataSentics, jako možné rozšíření jednoho z jejich produktů ShelfInspector. Ten zpracovává snímky regálů se zbožím pomocí natřenovaných neuronových sítí. Následně je pak díky tomu schopen poskytnout informace jako je počet a pozice konkrétního produktu, ale také jeho současnou cenu v daném regálu. V současné době jsou používány jednotlivě nafocené snímky, a právě proto je cílem mé práce prozkoumat možnosti (a omezení) analýzy regálů z videa pomocí metod registrace obrazu.

Kapitola 1

Registrace obrazu

1.0.1 co to je

Registrace obrazu je proces kombinování dvou a více snímků pořízených z odlišných pohledů, času nebo zdrojů do jednoho současněho systému [2]. Je například často používaná při zpracovávání dat ze satelitů, v počítačovém vidění nebo medicíně.

Registraci lze rozdělit do čtyř kroků:

1. detekce příznaků
2. korespondence příznaků
3. odhad transformace
4. transformace snímku

Resampling bude vynechán, neboť pro potřeby této práce si vystačíme s jednoduchým přeložením jednoho snímku přes druhý.

1.1 Příznaky a jejich detekce

Příznaky lze označit jako významné body obrázku, jako jednoduchý příklad lze uvést hrany nebo okraj objektu zobrazovaného na snímku. Je žádoucí, aby byl invariantní, nebo alespoň dostatečně robustní vůči ... jako rotace, škálování, osvětlení, rozmazání, šumu či změny v úhlu pohledu. Tuto schopnost nazveme opakovatelností(repeatability), tedy schopnost (daného detektoru) najít stejný bod na snímcích pořízených za různých podmínek.

Kromě samotné polohy bodu jsou ukládány tzv. deskriptory, které obsahují informace o intenzitě? bodu a jeho okolí.

V této práci byl jako detektor použit SURF[...](popis fungování)

1.2 Korespondence příznaků

Následně je spočítána vzdálenost mezi deskriptory a sobě nejbližší body prohlášeny za páry.
Bruteforce vs kNN(exhaustive vs approximate)

1.3 Odhad transformace

1.3.1 RANSAC

Je náhodně vzato n namatchovaných párů. Čím je jich méně, tím lépe, ale jsou třeba alespoň 4 páry pro projektivní a 3 pro affiní transformaci. Pomocí nich je odhadnut model transformace. Následně jsou všechny ostatní body ztransformovány podle této matice a ty které se nachází dostatečně blízko svému původnímu bodu jsou prohlášeny za inliers, ostatní za outliers. Model transformace s největším počtem inliers je zvolen jako výsledný.

1.3.2 Afinní a projektivní transformace

(definice hom.souradnice) Nechť je souradni bodu $x=(x_1,x_2)$. Pak jeho homogenní souřadnici je $x=(x_1,x_2,1)$. Bod $y=(y_1,y_2,y_3)$ prevede do R^2 jako $y=(y_1/y_3,y_2/y_3)$. $(x_1,x_2,0)$ značí že je bod v nekonečnu

Výsledná matice je lineární transformací, která je pro naše potřeby buď projektivní nebo affiní.

affiní, projektivní

co se zachovává při affiní, projektivní?

popis pomocí nehomogenních souradnic

1.4 transformace(pipelina?)

Abychom předešli vzniku děr a problémů s neceločíselnými souřadnicími při transformaci bodu zdrojového obrazu, používáme zpětnou metodu. Ta spočívá v procházení bodů warpovaného obrazu a určení zdrojového bodu pro daný bod pomocí zpětné transformace.

Kapitola 2

Výběr snímků z videa

2.1 focus measure

[<https://www.pyimagesearch.com/2015/09/07/blur-detection-with-opencv/>]

Nejkvalitnější snímky z videa jsou ty s nejmenším rozmazáním. Méně rozmazaný snímek zároveň zřejmě bude mít výraznější hrany než ten více rozmazaný. Hrany detekujeme pomocí konvoluce Laplaceova operátoru s vybraným snímkem. Výsledný (absolutní) průměr této matice nám bude sloužit jako nástroj k porovnávání rozmazání mezi snímky. Tato metoda není úplně přesná, protože se scéna na snímcích konstantě mění, nicméně se nemění tak dramaticky, aby nebyla použitelná.

2.2 Velikost okna

Předpokládáme rychlosť kamery 20 cm/s a frame rate 30FPS , tedy posun 2/3 cm/frame. Chceme aby se dva snímky překrývali alespon ze 1/3 (při menším překryvu by počítání tr. mat mohlo být obtížné/nepřesné) a zároveň se nepřerývali více než ze 1/2 (minimalizace počtu snímků). Předpokládáme, že snímek zobrazuje vertikálně 1 m ve skutečnosti. Pak se dostaváme na hodnotu okna 37-75 framů.

Nápad : Počítání rychlosti kamery. 2 kola

1. Vyberu dostatečně ostré snímky (jen na začátku/ celé video?). Možná budu potřebovat pevné okno?
2. Spočítám si projektivní tr. matice.
3. Pomocí velikosti posunu tr. matice a časovou vzdáleností spočítám průměrnou rychlosť.
4. Přes známou rychlosť spočítám přesnější okno

100 frameu uprostred-spočítam rychlosť z toho dynamicke zvetsivano/zmensovani okna podle predchozihho-lepsi

2.3 Error threshold

Pro odhad dobré transformace použijeme následující metodu: Označíme smer. odchylku intenzity snímků jako errThresh. Tato hodnota nám bude sloužit jako prah při porovnávání s chybou transformace. Ta je spočítána jako $\text{sum}(\text{abs}(I1-I2))$ kde I1,I2 jsou překrývající se části snímků. Pokud se chyba žádného snímků nevejde pod ErrThresh, je vybrán snímek s nejmenší chybou transformace v daném okně.

$$\text{mean} = \text{sum}(x)/\text{length}(x)$$

$$\text{variance} = \text{sum}((x - \text{mean}(x)).^2) / (\text{length}(x) - 1)$$

$$\text{std} = \text{sqrt}(\text{variance})$$

2.4 pipeline

Pro všechny snímky ve videu je spočítána focus measure. Všechny ostatní práce se snímky je postupně dělána ve vyhrazených oknech, což je soubor snímků vzdálený od předchozího vybraného snímku minimální a maximální vzdáleností (vzdáleností je myšleno časové pořadí snímků). Pro první průchod je jednoduše vybrán snímek s největší fm a spočítán ErrThresh pomocí něj. Zbytek cyklu probíhá takto:

1. Všechny snímky okna jsou v sestupném pořadí seřazeny podle jejich focus measure.
2. Snímky jsou jeden po druhém registrovány na vybraný snímek z předchozího okna, dokud není nalezen snímek s chybou transformace menší než ErrThresh. Pokud ani jeden není menší, je vybrán snímek s nejmenší chybou tr.
3. Tento snímek je přidán do konečného výběru snímků

2.5 Registrace panoramatu

[<https://www.mathworks.com/help/vision/ug/feature-based-panoramic-image-stitching.html>]

2.5.1 computeTforms

Transfromační matice jsou počítány následující způsobem: Snímek je převeden do černobílý (1 vrstva místo 3 u RGB) a jsou naněm detekovány body pomocí SURF detektora. Z těchto bodů jsou vybrány příznaky, které jsou následně spárovány s příznaky z předchozího snímku. Tyto 2 seznamy jsou poté použity funkcí estimateGeometricTransform, která spočítá odpovídající tr. matici. Pro potřeby panoramatu je použita affiní transformace, protože u projektivní dochází při součinu matic ke velké chybě.

Pokud chceme, aby referenčním snímkem v panoramu byl jiný než první, je potřeba je přepočítat pomocí tforms(j).T = tforms(j).T * Tinv.T; kde Tinv je inverze ref. snímku.

2.5.2 computeLimits

Je třeba znát rozměry výsledného panoramatu, čehož docílíme transformací 4 bodů okrajů u všech snímku a následným výběrem těch nejzaších. V tomto kroce je dobré ověřit velikost pro výsledné panorama, pokud je příliš velká, evidentně někde nastala chyba.

2.5.3 createPanorama

Díky znalosti rozměrů panoramatu si můžeme vytvořit konečný souřadný systém, do kterého nejdříve vložíme masku (opět 4 kraje původního snímku) a následně do místa masky vložíme transformovaný snímek. (Masku pak můžeme využít na zobrazení vizualizaci poskládání všech snímků)

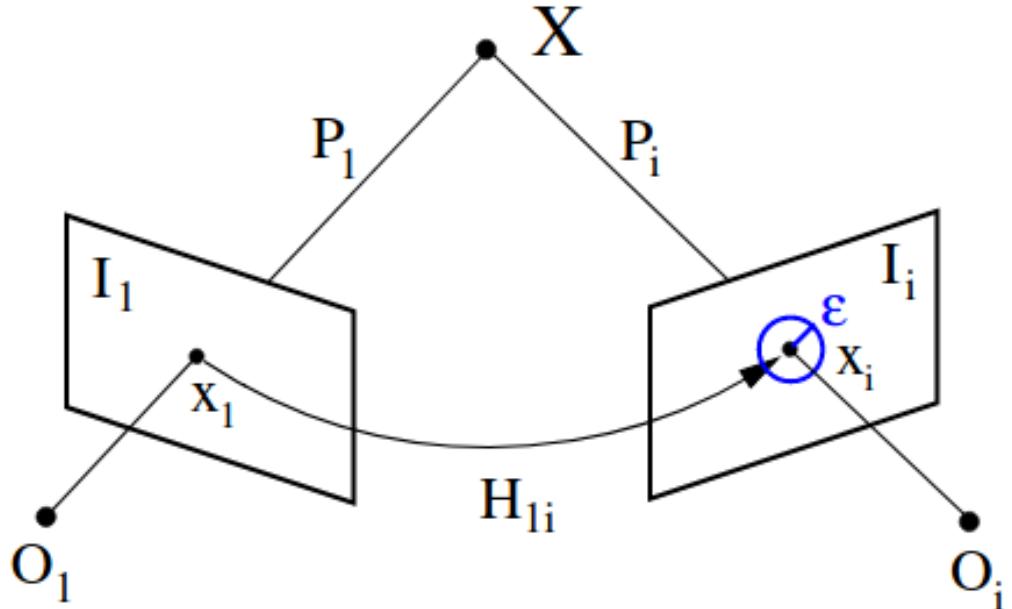
2.5.4 Omezení

Jak bylo výše uvedeno, u dlouhých panoramat s jakými pracujeme my, je třeba použít affiní, či podobnostní transformaci, neboť při součinu projektivních transformací dochází ke stále větší a větší chybě, a při více než 5-7 snímcích ani nelze žádné panorama vytvořit.

Kapitola 3

Experimenty

3.1 Opakovatelnost(repeatability)



Obrázek 3.1: Ilustrace principu opakovatelnosti

Opakovatelnost je podle [1] definována jako schopnost detektovat příznaky nezávisle na změnách ve snímacích podmínkách, jako je třeba posun kamery nebo zhoršení kvality obrazu.

Míra opakovatelnosti (repeatability rate) jako poměr stejných bodů vyskytujících se na obou snímcích vzhledem k celkovém počtu detekovaných bodů. Z detekovaných bodů vybíráme pouze ty, které jsou součástí scény vyskytujících se na obou snímcích.

Je důležité také vzít v úvahu nepřesnost detekce. Korespondující bod se nebude nacházet přesně daném místě určeném maticí, ale v jeho eps-okolí. eps-reapeatabilitu definujeme jako:

$$R_i(\varepsilon) = \{(x_i, x_j) \mid \text{dist}(H_{ij}x_i, x_j) < \varepsilon\} \quad (3.1)$$

Počet detekovaných bodů ve společné části snímku nebude stejný, proto vybereme menší z nich, abychom zajistili existenci bodu z eps-reapeatability. Repeatability rate je tedy definováno následovně:

$$r_i(\varepsilon) = \frac{|R_i(\varepsilon)|}{\min(n_i, n_j)} \quad (3.2)$$

My jako eps-repeatability bude používat inliers získané při odhadu tr. matice RANSAC metodou. Do jmenovatele pak dáme počet korespondujících párů, které leží ve společné části snímku.

Epsilon je třeba zvolit tak, aby pravděpodobnost, že se 2 náhodné body nachází ve svém eps-okolí velmi malá. Při rozlišení snímku 1920*1080 a počtu detekovaných bodů pohybujících se okolo 2000 až 200 bude průměrná vzdálenost bodu od sebe mnohem větší, než je $\text{eps}=10$ (většinou se používá 1.5?).

Dalším sledovanou hodnotou bude průměrná lokalizační chyba(localization error), což je průměrný vzdálenost bodu od jeho predikované pozice, tedy:

$$\text{loc} = \frac{\sum_{n=1}^N \|\psi_n - \hat{\psi}_n\|}{N} \quad (3.3)$$

3.2 Vliv rozmazání na registraci

Pro sledování vlivu rozmazání na kvalitu registrace použijeme dvojice sousedících snímků, které uměle rozmažeme aplikací motion-blur filtru náhodné orientace v rozsahu $<0,190>$ a pevně zvolené délky v rozsahu $<1,50>$, kdy je postupně přidávana po inkrementech 5(pixelů?). Z jedné dvojice tak získáme 11 (12 s originálními) párů rozmazaných snímků. U nich pak najdeme příznaky, namatchujeme je a odhadneme homografní matici. Pro E v Eps-repeatabilite dosadíme 10. Takte získáme repeatability rate.

Takto projdeme celý daný set snímků. Tady je graf setu 1:

Na vertikální ose je repeatability rate a jeho rozptyl, na horizontální délka filtru. Jak lze vidět z grafu, do cca 30 pixelu je kvalita registrace poměrně spolehlivá, pak ale začne rychle klesat.

3.3 vliv artefaktů

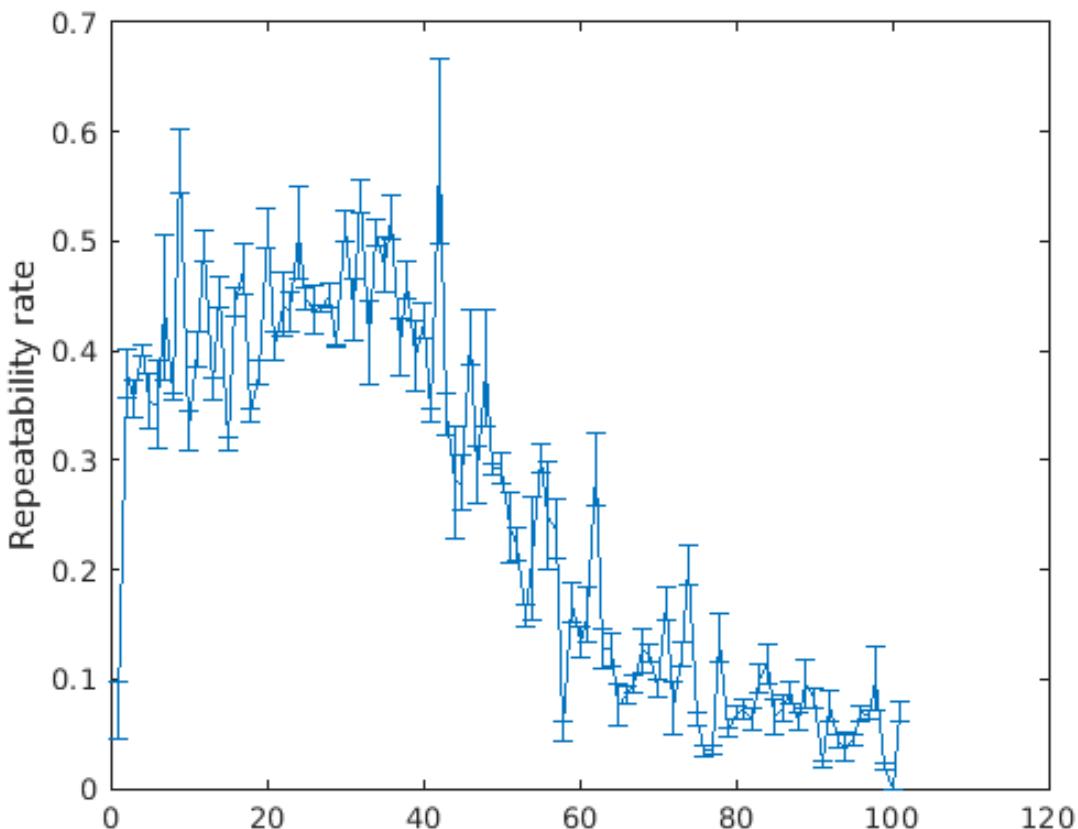
Sledování vlivu zmenšování datového toku kvalitu registrace. Plánovaný postup:

1. Vygeneruju set videí se různým datovým tokem
2. Problém s výběrem snímku (od nějakého to už nezvládne vybrat snímek)-> řešení: vybírám předem dané snímky (znám čísla z originálu)
3. Dále postup stejný jako u bluru (detekce příznaků, matching, estimateGeometricTransform,etc...))

3.4 Detekce cenovek

V této části si ukážeme využití výše zmíněných metod na problému počítání cenovek. V podstatě jde o schopnost detekce stejného objektu na více snímcích a následném rozdělení tak, aby každý snímek měl "své" cenovky. Pro toto je využití vlastností transfromačních matic ideální.

Pro samotnou detekci cenovek byla využita typ site YOLOv5[<https://github.com/ultralytics/yolov5>] natrénována na cenovky od firmy DataSentics. Ta vyniká především pro svou rychlosť i malou velikost, čímž je vhodná pro využití v mobilních zařízeních. Vstupem je snímek a výstupem list souřadnic bounding boxů.



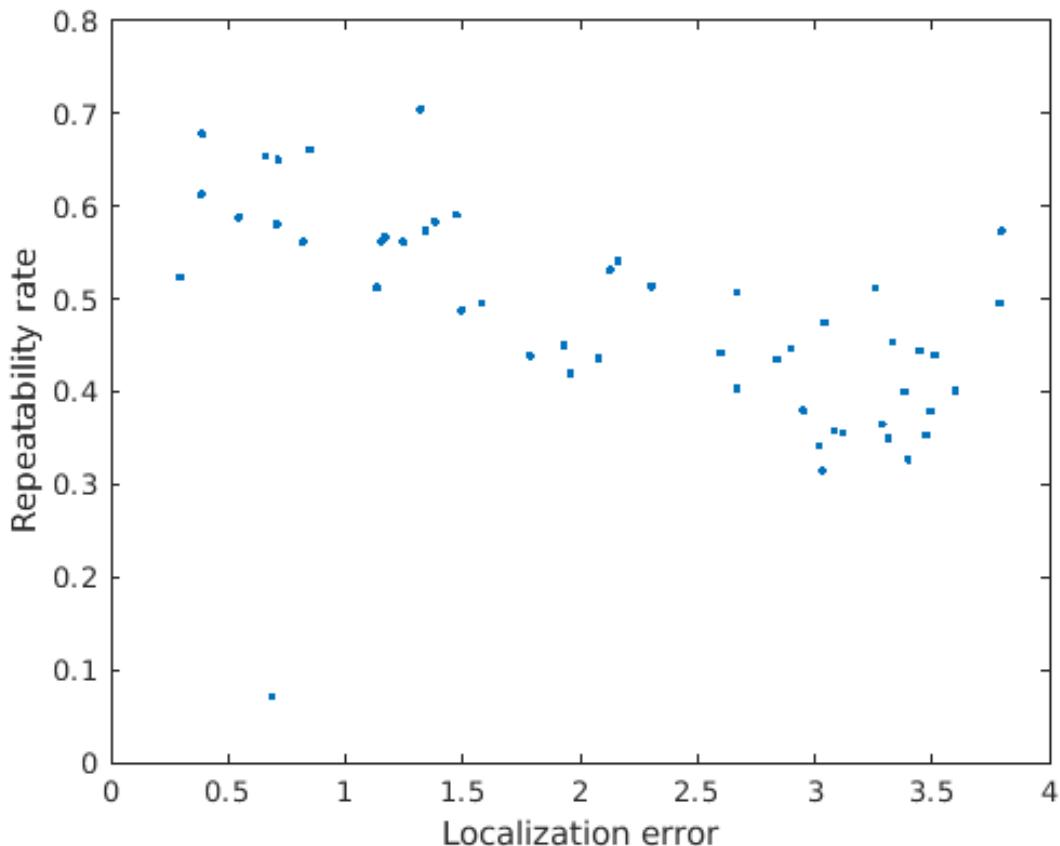
Obrázek 3.2: Zavislost rep. na délce rozmazání

Snímky jsou získány z videa postupem výše uvedeném. Mezi těmito snímky je spočítány matice projektivní transformace a je také (odděleně) vygenerováno panorama pomocí affiní transformace. Toto panoram slouží k ručnímu sčítání cenovek.

V každém cyklu jsou vzaty 3 po sobě jsoucí snímky, v nichž jsou následně detekovány cenovky. První a třetí bboxy jsou převedeny transformací do souřadného systému druhého snímku. Poté jsou přes Python knihovnu Shapely[] souřadnice bboxů převedeny do polygonů pro snadnější manipulaci.

Polygony prvního a druhého snímku jsou dávány přes sebe a pokud je nalezen překryv, polygon je odstraněn z listu ve 3. a zůstává pouze ve 1. Následně jsou podobným způsobem kontrolovány polygony prvního a druhého a třetího a druhého. Při nalezení překryvu ale o tom, ze kterého listu bude polygon smazán rozhoduje velikost obsahu cenovky.

Po prvním cyklu neprochází již detekované snímky sítí znova, místo toho jsou brány seznamy z předchozích 2 cyklů. Díky tomu nedochází k duplikaci cenovek na různých snímcích. Mohlo by v nějakých případech dojít k duplikaci, pokud by se objekt objevil na 4 a více snímcích po sobě, nicméně k tomu v podstatě nedochází, vzhledem k tomu že snímky jsou vybrány z videa, tak, aby neměly tak velký překryv. Nehledě na to, že při součinu projektivních transformačních matic se rychle kumuluje chyba po několika cyklech by se bboxy přestaly překrývat.



Obrázek 3.3: Zavislost rep. na délce rozmazání

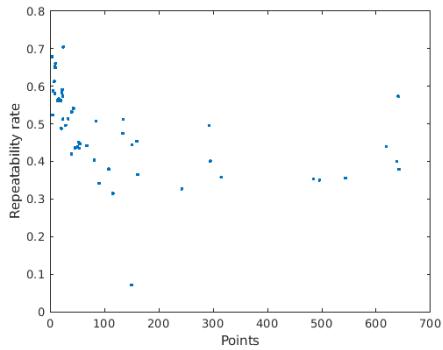
Číslo setu	1	2	3	4	5
Sečteno ručně	188	190	xxx	xxx	xxx
Sečteno programem	102	96	xxx	xxx	xxx

Tabulka 3.1: TABLETABLETABLE.

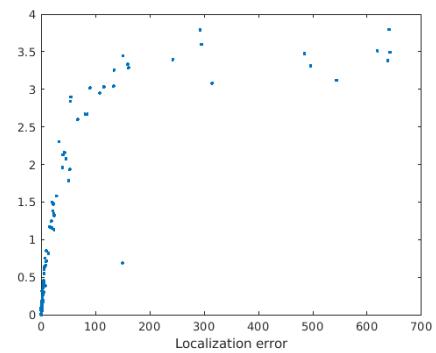
3.5 Omezení

Sít má problém s cenovkami velmi blízko u sebe a najde neexistující cenovky. Také má problém správně detekovat cenovky vyloženě přes sebe a neobvyklých formátů (elektronické, moc dlouhé)

3.6 Výsledky



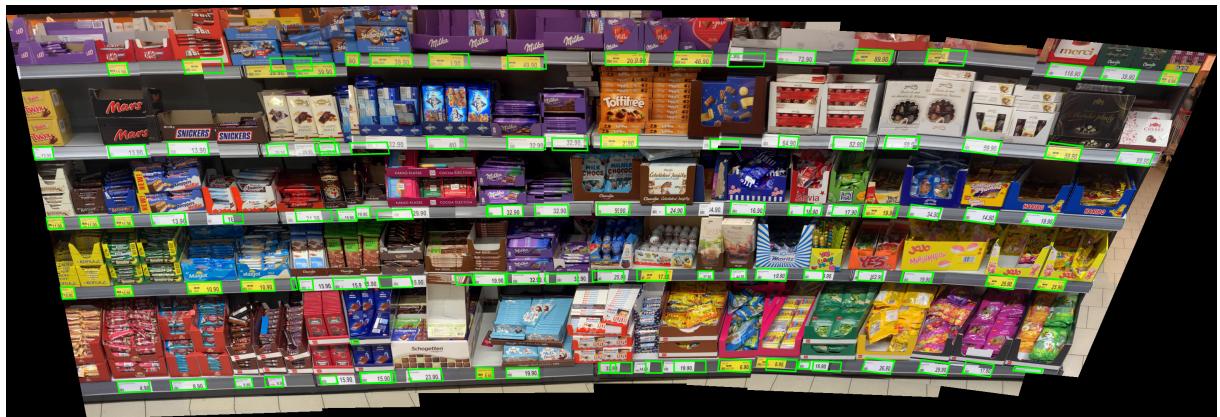
Obrázek 3.4: závislost rep.rate na počtu bodu



Obrázek 3.5: aaaa



Obrázek 3.6: Zobrazení cenovek na panoramatu z vid30.



Obrázek 3.7: Cenovky na panoramatu z vid34

Závěr

Text závěru....

Bibliografie

- [1] Cordelia Schmid, Roger Mohr a Christian Bauckhage. “Evaluation of Interest Point Detectors”. In: *International Journal of Computer Vision* 37 (srp. 2000), s. 151–172. doi: [10.1023/A:1008199403446](https://doi.org/10.1023/A:1008199403446).
- [2] Barbara Zitová a Jan Flusser. “Image Registration Methods: A Survey”. In: *Image and Vision Computing* 21 (říj. 2003), s. 977–1000. doi: [10.1016/S0262-8856\(03\)00137-9](https://doi.org/10.1016/S0262-8856(03)00137-9).

Seznam obrázků

3.1	Ilustrace principu opakovatelnosti	13
3.2	Zavislost rep. na délce rozmazání	15
3.3	Zavislost rep. na délce rozmazání	16
3.4	závislost rep.rate na počtu bodu	17
3.5	aaaa	17
3.6	Zobrazení cenovek na panoramatu z vid30.	17
3.7	Cenovky na panoramatu z vid34	17

Seznam tabulek

3.1 TABLETABLETABLE	17
-------------------------------	----