

Dokumentace - ZPO 2021

Segmentace obrazu pomocí optického toku

Autor: Lukáš Dobiš Login: xdobis01 Dátum: 26.4.2021

1 Úvod

Ako projekt bolo vybrané zadanie na segmentáciu obrazu pomocou optického toku. Optický tok sa používa na ohodnotenie pohybu na úrovni pixelov medzi jednotlivými obrazmi z videa. Tento projekt rozoberá optický tok, spôsoby jeho získania a implementuje operátor vlastného návrhu na jeho spracovanie na segmentačný obraz.

2 Zadanie

Cieľom projektu je vytvorenie metódy ktorá využije detekcie optického toku na segmentáciu obrazu na oblasti s konštantnou zmenou smeru.

- Vstupom metódy je video, ktorej medzi výsledkom bude optický tok medzi dvomi snímkami.
- Optický tok je zpracovaný nejakým operátorom, ktorý oblasti s podobným smerom a veľkosťou vektorov spojí do segmentov. Výsledkom je segmentovaný obraz.
- Vyskúšajte rôzne metódy optického toku a vyhodnoťte najvhodnejšiu z hľadiska kvality a rýchlosťi.

2.1 Upresnené zadanie

V projekte som v prieskume metód na segmentáciu pomocou optického toku zistil, že väčšina zo súčasných state-of-art metód je založená na konvolučných sieťach [4]. Návrh, implementácia a návrh takejto siete je mimo hlavného obsahu predmetu ZPO, a použitá knižnica opencv na spracovanie videa [1], obsahuje iba Farneback metódu pre výpočet optického toku v celom obraze [2]. Rozhodol som sa preto v implementácii riešenia zamerať na návrh vlastného operátoru na získanie segmentácie z optického toku Farneback metódy, a vyhodnotenie jeho efektivity.

3 Metódy získania optického toku

Nasledujúca časť obecne popisuje základný princip najznámejších differenciálnych metód odhadu optického toku. Metódy založené na fázovej dekorelácii, diskrétnej optimalizácii a blokoch sú vynechané.

3.1 Horn-Shunck metóda

Horn-Shunck metóda je klasickým algoritmom na odhad optického toku, založenom na hladkosti toku v celej rovine obrazu [3]. Snaží sa teda minimalizovať nehladké časti toku a hľadá riešenie s najhladším možným tokom. Algoritmus začína s počiatočným hrubým odhadom toku a následne ho iteratívne zjemňuje do podoby finálneho riešenia. Ciel je definovaný cez objektívnu funkciu v podobe globálnej energie, ktorú je potrebné minimalizovať. Nech $p = (x, y)$ je obraz a z ktorého sa bude dopočítavať pole optického toku $w(p) = (u(p), v(p), 1)$, kde $u(p)$ a $v(p)$ sú horizontálne a vertikálne zložky tohto toku. Potom výsledná objektívna funkcia pre spojity priestor má podobu rovnice 1, kde ∇ je gradient, λ regularizačná váha a I_1, I_2 sú dva obrazy medzi ktorými sa počíta tok. Metóda iteráčne dopočítava úpravy dw pre pole w pre minimalizáciu tejto funkcie. Metóda je výpočetne náročná.

$$E(u, v) = \int |I_2(p + w) - I_1(p)|^2 + \lambda(|\nabla u|^2 + |\nabla v|^2), dp \quad (1)$$

3.2 Lucas-Kanade metóda

Lucas-Kanade metóda je menej výpočetne náročná a jednoduchšia metóda [6], ktorá odhaduje tok tým, že sleduje pohyb niekoľkých zaujímavých bodov medzi po sebe nasledujúcich obrazoch. Táto metóda predpokladá že pohyb medzi obrazmi není príliš veľký a zobrazovaná scéna v obraze obsahuje objekty, ktorých povrch nemá prudké zmeny intenzity. Namiesto hľadania pridruženého pixelu v druhom obraze sa vystačí iba s odhadom jeho pohybu na základe zmeny lokálnej intenzity. Body sú zvolené Shi-Tomasovým algoritmom a majú pre svoje okolné pixely vytvorené rovnice pre ktoré sa hľadá posun v čase pre smer x a y osí. Pre zvýšenie detekčnej citlivosti sa využíva tzv. obrazová pyramída ktorá spočíta tok z viacerých rozlišení obrazu.

3.3 FarneBack metóda

Farneback metóda je podobná Lucas-Kanade metóde v použití bodov a ich okolia. Metóda okolie bodov cez kvadratické polynomiály rozšírené polynomiálnou expanziou. A z koeficientov získaných polynomiálov následne odhaduje posunutie medzi okoliami ako optický tok [2].

4 Vlastný operátor na spracovanie toku

Optický tok obsiahnutý v šedotónovom obraze je značne náchylný na detekciu šumu v podobe zmeny osvetlenia v čase. Cieľom operátora bude odstrániť tento šum a súčasne minimálne potlačiť skutočne pohybujúce sa objekty. Optický tok je prevedený z vektorovej matice do polárnej sústavy v podobe matice veľkosti vektoru a jeho uhlu. V tejto podobe je veľkosť vektoru spracovaná nasledujúcim spôsobom. Je binárne naprahovaná na odstránenie najmenšej zložky šumu prítomnej v celom obraze a postupne sa aplikujú 3 morfologické operátory so štruktúrnym elementom štvorca. Ako prvý sa aplikuje dilatácia o veľkosťi 3 ako predpríprava na eróziu, pre zrobustnenie podstatnej informácie v obraze. Nasledovná erózia má veľkosť operátora až 12, čím sa potlačí väčšina šumu v obraze a z pohybujúcich zložiek toku zostanú iba čiastočné kostry pôvodných pohybujúcich sa objektov. Objem toku objektov je čiastočne späť rekonštruovaný dilatáciou o veľkosťi 10, predpokladá sa, že šum ktorý sa týmto spôsobom zvýrazní bol už väčšinovo potlačený eróziou.

5 Implementácia

Projekt bol implementovaný v jazyku C++ s použitím normy C++11. Na manipuláciu s obrazovými dátami a výpočet optického toku bol použitý modul opencv [1]. Výsledný program má názov segmentVideo.

5.1 Obsluha programu

Na preklad a správu závislostí bol použitý CMake program. Program má ako posledný vstup cestu k videu na segmentáciu, zvyšnými vstupmi sú voliteľné prepínače a ich hodnoty. Prepínač -s so vstupom cesty k novému súboru s koncovkov .avi, spustí ukladanie vypočítanej segmentácie do daného súboru. Prepínač -c spolu s vstupom čísla v dátovom type float, určuje zmenu veľkosti obrazu pred výpočtom optického toku, čo ovplyvňuje množstvo šumu a pohybu vo výslednom toku a samotnú rýchlosť výpočtu. Obdobne platí pre prepínač -f, ktorý ale ovplyvňuje iba finálnu veľkosť obrazu, bez dopadu na podobu toku. Posledný prepínač -b slúži na zobrazenie/ukladanie iba veľkosti optického toku, bez jeho uhlovej zložky na určenie smeru pohybu v obraze. Príklad spustenia programu s využitím všetkých voliteľných prepínačov je nasledovný 1 .

```
segmentVideo -s ./path/segmentation.avi -c 0.6 -f 0.75 -b ./data/video.avi
```

Obr. 1: Príklad spustenia programu

5.2 Výpočet optického toku a segmentácia

Po načítaní obrazového páru, je každý obraz zmenšený na 0.6 pôvodnej veľkosti, ak nie je nastavená –c prepínačom iná hodnota. Tým sa zníži náročnosť výpočtu optického toku Farnebackovou metódou. Vstupné parametre pre Farnebackovou metódu ako počet iterácií, veľkosť a filter okna, počet úrovni pyramídy a ďalšie boli pre empiricky optimalizované pre najmenej zašumený obraz bez poškodenia pohybujúcej sa zložky toku. Pre zväčšenie pohybovej zložky a zmenšenie šumu sa každý druhý obraz vynechá, čo znižuje výsledný kmitočet segmentačného videa na polovicu. Optický tok je následne prevedený do polárnej sústavy súradníc. V tejto podobe je na obraz aplikovaný operátor z predchádzajúceho paragrafu 4. Následne je do obrazu veľkosť toku v polárnej sústave, pridaná uhlová zložka prevedením do HSV formátu a z neho do výsledného BGR formátu. Použitím –b prepínaču sa tento krok dá nahradíť prevedením veľkosti do výsledného šedotónového obrazu. Výsledný obraz je transformovaný na pôvodnú veľkosť, ak neboli použití –f prepínač na určenie novej veľkosti.

5.3 Dáta

Ako video dátá vhodné na segmentáciu bolo použité video stiahnuté z webovej stránky Youtube na ktoré sa nevzťahujú autorské práva [5]. Video má rozlíšenie 1920x1080 a 60 FPS so štandardnou 8-bitovou hĺbkou. Zobrazenou scénou na videu je rušná premávka v strede mesta, s autami, chodcami a skútrami. Tým splňa požiadavky vhodnej úlohy pre segmentáciu optickým tokom.

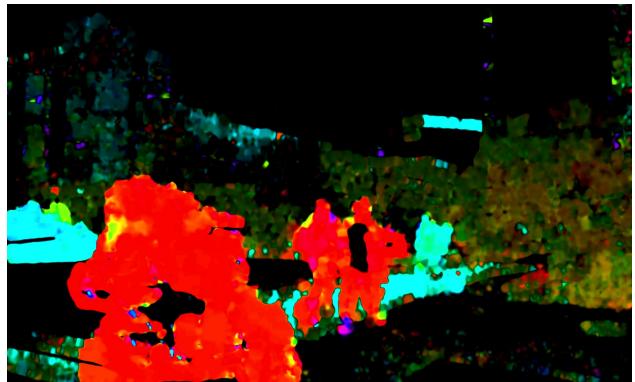
6 Výsledky

6.1 Dopočítaný optický tok

Na obrázkoch 2 je obraz z originálneho videa a na obrázku 3 je výsledný optický tok získaný Farnebackovou metódou implementovanou v opencv, po prevedení do polárnej sústavy a farebnej schémy RGB. Obraz použitý na výpočet optického toku bol zmenšený na 0.85 pôvodnej veľkosti. Modrú zložku obrazu tvorí pohyb smerom k pozorovateľovi, červenú zložku obrazu zas pohyb od pozorovateľa a zelenú zložku tvorí pohyb do strán. Z obrázkov je možné zhodnotiť, že tok zachytáva vo videu veľmi veľa rušivých elementov, ktoré sa v skutočnosti nehýbu a sú spôsobené zmenou svetla. Naopak pohyb niektorých druhov povrchu nie je zachytený, ako napríklad na pozorovatela kolmý povrch auta alebo povrch čiernej farby.



Obr. 2: Pôvodný obraz, rozlíšenie 1920x1080

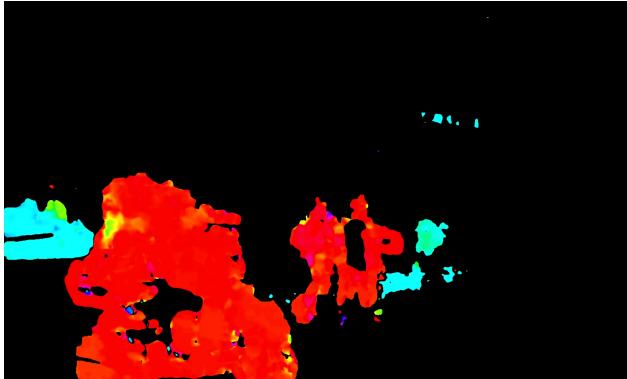


Obr. 3: Optický tok zo zmenšeného pôvodného obrazu

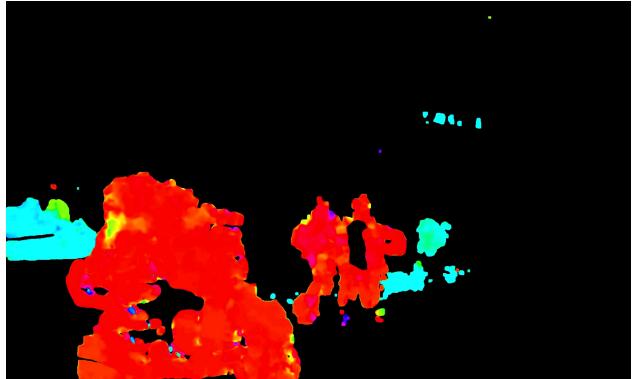
6.2 Aplikácia vlastného operátoru

Na obrázku 4 je vidieť výsledok operácie binárneho prahovania na optický tok z obrázku 3, je vidno že väčšina šumu bola potlačená a zostala hlavne zložka toku s pohybujúcimi objektami. Svetlomodrý tok osoby v zadnej časti, ale splynul so zvyšným šumom z prechodu chodcov a obraz stále obsahuje tyč so semafórom. Aplikácia prvej dilatácie 5 zvýšila objem všetkých objektov a pomohla vytvoriť plňšie segmenty. Nasledujúca rozsiahla

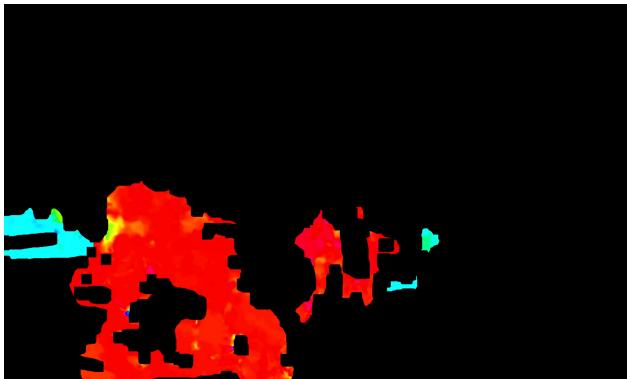
erózia 6 potlačila zvyšný menší šum, vrátane tyče s semafórom. Výnimkov je zachovaný šum z prechodu chodcov. Po druhej dilatácii vzniká výsledný segmentačný obraz 7, dilatácia navrátila objem objektov stratený pri predchádzajúcej erózii. Výsledný segmentovaný obraz obsahuje pohybujúce objekty s potlačeným šumom za cenu detailov obrysu pohybujúcich sa objektov.



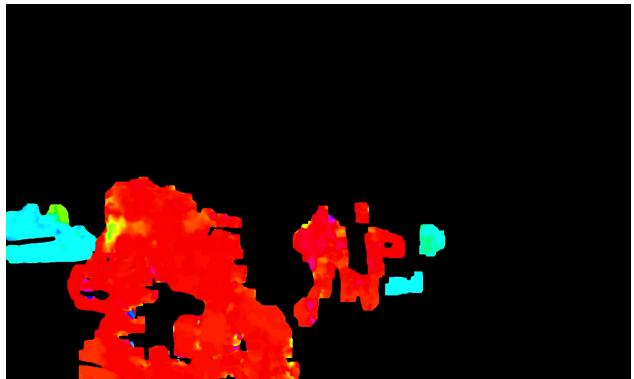
Obr. 4: Aplikácia binárneho prahovania



Obr. 5: Aplikácia prvej dilatácie, veľkosť 3



Obr. 6: Aplikácia erózie, veľkosť 12



Obr. 7: Aplikácia druhej dilatácie, veľkosť 10

6.3 Dopad zmeny veľkosti výpočetného obrazu

Na pozorovanie dopadu zmeny boli vybrané 2 rôzne scény, ktoré mali dopočítaný tok z 0.4 a 0.8 veľkosti pôvodného obrazu s rozlíšením 1980x1024. Ďalej bola vyhodnotená závislosť priemerného času na výpočet jedného segmentačného obrazu na rozdielnych velikostach obrazu v tabuľke 1. Na výpočet pre porovnanie priemerných časov bolo použité CPU AMD Ryzen 5 2600. Na obrázkoch 8 a 9 zo scény 1 je možné pozorovať, že pre menšiu veľkosť sa stratí optický tok vzdialenejších objektov a detaily z obrysu auta. To isté platí aj v scéne 2 na obrázkoch 11 a 11. Ale menšia veľkosť má zas výhody v podobe menšieho šumu oproti vyšším veľkostiam a značne zvýšenej rychlosťi, ako sa dá vyvodiť z hodnôt tabuľky 1.

Výpočetná veľkosť obrazu	396x205	792x410	1188x615	1584x820	1980x1024
Priemerný čas výpočtu [s]	0.04334	0.1137	0.2569	0.4243	0.6782

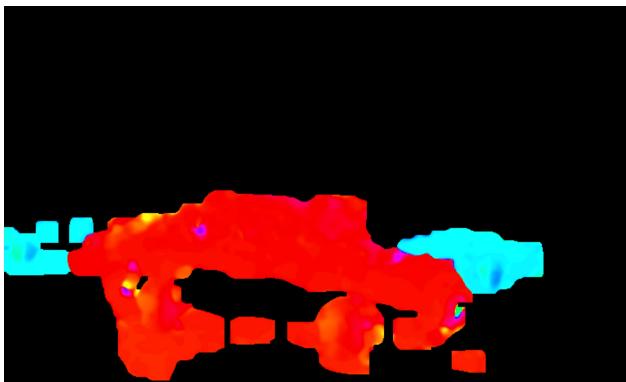
Tabuľka 1: Tabuľka závislosti hodnôt priemerného času na výpočet 1 segmentačného obrazu na jeho rozlíšení



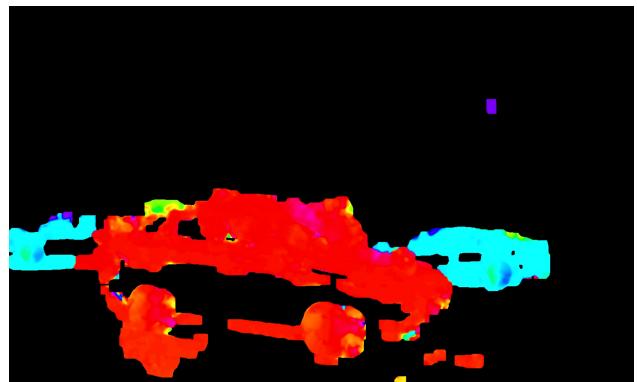
Obr. 8: Segmentačný obraz zo scény 1, veľkosť 0.4



Obr. 9: Segmentačný obraz zo scény 1, veľkosť 0.8



Obr. 10: Segmentačný obraz zo scény 2, veľkosť 0.4



Obr. 11: Segmentačný obraz zo scény 2, veľkosť 0.8

7 Záver

V tejto technickej správe projektu bol vypracovaný rozbor optických tokov a navrhnutý operátor na získanie segmentačného obrazu. Bolo implementované riešenie, ktoré vypočítava optický tok z videa Farnebackovou metódou a následne z neho segmentuje obraz skrz aplikáciu navrhnutého operátora. V dosiahnutých výsledkoch je diskutovaný dopad zmeny veľkosti obrazu pri výpočte toku na samotnú podobu toku a rýchlosť jeho výpočtu.

Referencie

- [1] Bradski, G. and Kaehler, A., 2008. *Learning OpenCV: Computer vision with the OpenCV library*. Ö'Reilly Media, Inc.”.
- [2] Farnebäck, G., 2003, June. *Two-frame motion estimation based on polynomial expansion*. In Scandinavian conference on Image analysis (pp. 363-370). Springer, Berlin, Heidelberg.
- [3] Horn, B.K. and Schunck, B.G., 1981. *Determining optical flow*. Artificial intelligence, 17(1-3), pp.185-203.
- [4] Jonschkowski, Rico and Stone, Austin and Barron, Jonathan T and Gordon, Ariel and Konolige, Kurt and Angelova, Anelia. *What matters in unsupervised optical flow*. arXiv preprint arXiv:2006.04902, Springer, 2020.
- [5] G Korb Free City Street Footage - Free Video No Copyright. [Online] Zdroj: www.youtube.com
- [6] Lucas, B.D. and Kanade, T., 1981, April. *An iterative image registration technique with an application to stereo vision*.