SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

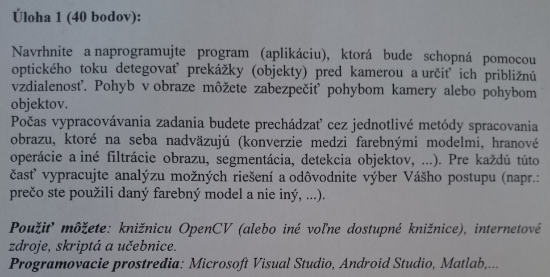
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Vizuálne systémy

Dokumentácia: ÚLOHA 1

**Bratislava 2017 Michal Pavlac**

**Zadanie:**



**Vypracovanie:**

Na začiatku sme si vyhľadali informácie o téme ktorú budeme spracovávať. Najprv sme sa zamerali na optický tok. Detekciu prekážok sme implementovali hlavne pomocou tutoriálov na internete. Využívali sme nasledovné funkcie:

*goodFeaturesToTrack* – nájdenie hrán najmä v oblasti ostrých vrcholov objektov v obrázku *cornerSubPix* – upresneni polohy rohov na obrázku

*calcOpticalFlowPyrLK* – kalkulácia optického toku pomocou iteratívnej Lucas-Kanade metódy s pyramídami

\*Parametre týchto funkcií sme si vyhľadali v dokumentácií opencv a vhodne nastavili

Na nasledujúcom obrázku môžeme vidieť ako vyzerá grafická interpretácia našej detekcie prekážok pomocou optického toku:



Ďalej sme sa zaoberali problémom určovania vzdialenosti predmetu známych rozmerov od kamery. Tak isto sme si vyhľadali na internete spôsoby, akými by tento problém bolo možné riešiť. Rozhodli sme sa na tento účel použiť Houghovu transformáciu, konkrétne pre určovanie kruhov. Odmerali sme si polomer predmetu, v našom prípade CDčka, ktoré malo polomer 6cm. Položili sme ho pred kameru do určitej známej vzdialenosti. Zistili sme si polomer kruhu (CD) v pixeloch. Vďaka týmto parametrom sme dokázali vypočítať ohniskovú vzdialenosť kamery. Vďaka tomu sme dostali poslednú potrebnú premennú do vzorca:

*Ohn. vzdialensoť = (polomer v pixeloch \* známa vzdialenosť od kamery v cm)/polomer v cm*

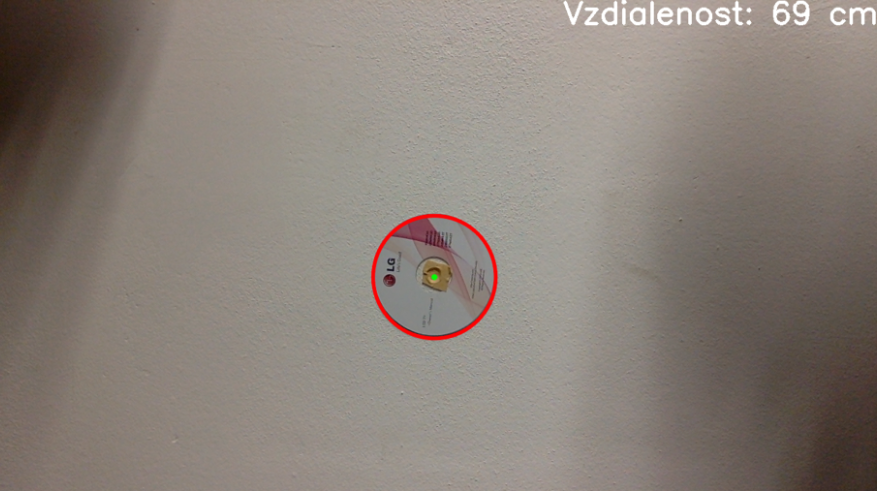
a dokážeme určovať vzdialenosť daného kruhového predmetu od kamery.

Použili sme tieto funkcie:

*houghCircles* – vstupom je obraz v tzv. gray modeli (čierno-biely) a výstupom je pole 3-zložkových vektorov, kde prvé 2 zložky sú súradnice stredu a tretia súradnica je polomer kruhu (všetko v pixeloch)

*medianBlur* – rozostrenie obrazu – zlepšovalo detekciu kruhov, znižovalo pravdepodobnosť detekcie nepravého kruhu

Na nasledujúcom obrázku môžeme vidieť ako vyzerá grafická interpretácia nášho určovania vzdialenosti:



Detekcia prekážok spolu s určením vzdialenosti vyzerá takto:

