

Počítačové videnie - BRISK, RANSAC

Ing. Viktor Kocur
viktor.kocur@fmph.uniba.sk

DAI FMFI UK

7.4.2020

Nájdenie bodov

Postup

Lokálne príznaky získame tak, že najprv nájdeme v obrázku kľúčové body. Teda miesta ktoré sú vhodné na hľadanie pri otočení a podobne. Takéto miesta môžu byť hrany, rohy, alebo bloby. Následne každému bodu spočítame deskriptor. Teda vektor popisujúci bod a jeho okolie. Ideálne chceme aby bol tento deskriptor invariantný na rôzne transformácie obrazu, resp. zmenu pozície bodu v reálnej scéne.

SIFT

V prednáške ste preberali metódu SIFT, ale tá je bohužiaľ patentovaná a tak v Matlabe natívne nieje. Je možné stiahnuť kód z Matlab central, alebo odinadiaľ. Kvôli možným problémom s rôznymi OS však budeme používať iné metódy.

Metódy dostupné v matlabe - detekcia kľúčových bodov

Detekcia kľúčových bodov

V Matlabe sú okrem iných dostupné detektory SURF, BRISK, FAST a Harissov detektor.

Matlab implementácia

Jednotlivé príkazy pre tieto dektory sú podobné: `detectSURFFeatures`, `detectBRISKFeatures`, `detectFASTFeatures` a `detectHarrisFeatures`. Tieto funkcie vracajú custom objekty s ktorými budeme ďalej pracovať.

Detekcia kľúčových bodov v Matlabe

Nájdenie a zobrazenie 20 najlepších kľúčových bodov

```
p = detectXXXXFeatures(I);  
imshow(I);  
hold on;  
plot(p.selectStrongest(20));
```

Úloha

Zobrazte si zaujímave body zo scene.pgm, box.pgm a book.pgm.
Pre rôzne detektory.

Extrakcia príznakov - deskriptorov

Extrakcia príznakov

V ďalšom kole priradíme každému kľúčovému bodu deskriptor. Ten nám pomôže nájsť tento bod v inom obrázku.

extractFeatures

`[f, valid_p] = extractFeatures(I, p)` - pre šedotónový obrázok `I` a objekt kľúčových bodov z predchádzajúceho kroku vráti objekt `f` obsahujúci deskriptory pre každý vhodný bod, `valid_p` bude obsahovať tie body z `p`, ktoré sa dali použiť. Príkaz automaticky vyberie metódu podľa typu bodov, ale môžeme ju zmeniť ak pridáme argument `'Method'` (pozrite si help). Bodý `p` môžu byť aj jednoduchá matica $N \times 2$, kde na každom riadku sú súradnice kľúčových bodov.

Párovanie bodov

Úloha

Pozrite si v objekte `f` aké deskriptory boli vygenerované.

Párovanie bodov

Ak máme dva obrázky môžeme v nich nájsť kľúčové body a potom ich spárovať na základe ich deskriptorov.

`matchFeatures`

`idxs = matchFeatures(f1, f2)` - vráti maticu `idxs`, ktorá na každom riadku obsahuje v prvom stĺpci index pre bod z `f1` a v druhom stĺpci index pre bod z `f2` pre jeden pár. `f1` a `f2` sú feature objekty z predchádzajúceho kroku. Párovanie si môžete zobrazit' napr. pomocou priloženého súboru `match.m`.

Hľadanie homografie

Homografia - transformácia

Hľadáme transformáciu medzi rovinou na hľadanom vzore a obrázkom kde sa tento objekt nachádza.

Ako nájsť homografiu?

Homografiu hľadáme tak, že minimalizujeme funkciu $\sum_i \text{res}(H, \vec{x}_i, \vec{x}'_i)$. Pre homografie platí že H je matica 3×3 .

Homogénne súradnice

Pre použitie matice homografie využívame tzv. homogénne súradnice. Každý bod v obraze reprezentujeme tromi číslami: x_h, y_h, z_h . Pre prechod na štandardné súradnice platí $x_s = \frac{x_h}{z_h}$ a $y_s = \frac{y_h}{z_h}$. V prípade že $z_h = 0$ ide o bod v nekonečne.

RANSAC

Implementácia

Implementáciu RANSAC-u si môžete stiahnuť zo stránky. Ak nechcete mať všetky súbory spolu so SIFT-om v jednom adresári, môžete si zložku ransac pridať do MATLABPATH.

RANSAC

Funkcia ktorú minimalizujeme

$$\sum_i res(H, \vec{x}_i, \vec{x}'_i) = \sum_i \rho(H\vec{x}_i, \vec{x}'_i) + \rho(\vec{x}_i, H^{-1}\vec{x}'_i).$$

`ransacfithomography`

`[H, inliers] = ransacfithomography(x1, x2, t)` - Pre spárované body `x1` a `x2` (môžu byť aj v štandardných súradniciach) a prah `t` (0.001 - 0.01) vráti maticu homografie `H` a zoznam indexov pre `x1` a `x2`, ktoré sú spárované správne.

Úloha

Úprava match

Upravte funkciu match, tak aby volala RANSAC a našla homografiu. Na pôvodnom obrázku zobrazte kde sa nachádza vzor. Zeleným vykreslite inliers a červenými nechajte ostaé ostatné. Otestujte pre obrázky scene.pgm s book.pgm a box.pgm.

