

2. Domača naloga

Robotika in računalniško zaznavanje, 30. november 2010

Detekcija krogov

Napišite funkcijo `circles`, ki v podani sivinski sliki z uporabo Houghove transformacije poišče kroge. Postopek naj bo približno tak:

- Sliko segmentirajte z uporabo funkcije `autothreshold`, ki ste jo napisali na prvih vajah. Če prve vaje niste opravili, lahko uporabite tudi funkcijo, ki je priložena tem navodilom (na koncu dokumenta).
- Očistite morebitni šum na binarni sliki z uporabo morfoloških operacij
- Z uporabo vgrajene funkcije `hough_circle` izračunajte akumulatorsko polje Houghove transformacije. Pri tem lahko predpostavite določen razpon za radij kroga.
- Iz akumulatorja izluščite N najboljših kandidatov.

Glava funkcije naj izgleda tako: `function [C] = circles(imgin, N)`, kjer je `imgin` vhodna sivinska slika, N število ekstremov, ter `C` matrika rezultatov (velikosti $N \times 3$).

Vizualizacija detekcij

Napišite funkcijo `showcircles`, ki prikaže podano barvno sliko ter nad njo izriše z uporabo funkcije `circles` najdene kroge. Za izris kroga si pomagajte s funkcijo `plot` (krog izrišete kot zaporedje črt).

Glava funkcije naj izgleda tako: `function [] = showcircles(imgin, N)`, kjer je `imgin` vhodna sivinska slika, N število ekstremov.

Momenti

Napišite funkcijo `cmoment`, ki za podano binarno sliko `imgin` ter parametra p in q izračuna (p,q) -ti *centralni* moment.

Pri tem bodite pozorni tudi na hitrost same implementacije (priporočeno je, da se izogibate eksplicitni uporabi zank).

Glava funkcije naj izgleda tako: `function [m] = cmoment(imgin, p, q)`.

Splošna navodila

- Nalogo morate oddati preko Učilnice najkasneje do **19. decembra 2010**.
- Pošljite zip datoteko, katere ime je vaša vpisna številka. Arhiv naj vsebuje Octave/Matlab združljive datoteke, ki imajo ime enako imenu ustrezne funkcije.
- V funkcijah lahko uporabite samo standardne funkcije iz paketov `image` in `statistics`. Če uporabite še kakšen drug *standarden* paket, mi to omenite v komentarjih ali v priloženi tekstovni datoteki. Seveda to ne pomeni, da lahko uporabite samo ovijete (morebiti obstoječo) funkcijo, ki implementira celoten zahtevani algoritem.
- **Koda mora biti dobro dokumentirana!** Za vsako *netrivialno* vrstico kode v komentarju

- povejte kakšen je njen pomen v algoritmu.
- Za morebitna vprašanja me lahko kontaktirate na luka.cehovin@fri.uni-lj.si ali na forumu na Učilnici.
- Pravočasno oddane naloge bodo trenutno na hitro pregledane glede izpolnjevanja osnovnih zahtev, navedenih zgoraj. Konec semestra bom vse zbrane naloge še malo bolj podrobno pregledal in tiste, pri katerih bo obstajal sum prepisovanja (ter nekaj naključnih izbrancev) povabil na obvezen zagovor.

Priloga: datoteka autothreshold.m

```
function [ imgout ] = autothreshold(imgin)
h = histogram(imgin, 256);
mine = 10000;
t = 0;
for i = 2:length(h)-1
    h1 = h(1:i+1);
    h2 = h(i:end);
    h1 = h1(h1 > 0);
    h2 = h2(h2 > 0);
    e1 = -sum(h1 .* log(h1));
    e2 = -sum(h2 .* log(h2));
    if (mine > abs(e1 - e2))
        mine = abs(e1 - e2);
        t = i;
    end;
end;
imgout = threshold(imgin, t);
```