

Vaja 4: Robovi, gradient slike in morfološke operacije

Janez Pers

Laboratorij za strojni vid
Fakulteta za elektrotehniko, Univerza v Ljubljani
e-mail: janez.pers@fe.uni-lj.si

Povzetek

V okviru te vaje se bomo spoznali z nekaterimi uveljavljenimi metodami obdelave slike, ki temeljijo na izločanju robov, ter si ogledali še postopke, ki jih pogosto uporabljamo na binarnih (dvoobarvnih) slikah.

1 Izločanje robov (20%)

Pri tej vaji boste uporabljali sliko, ki jo lahko snamete iz naslednjega naslova:

<http://vision.fe.uni-lj.si/classes/RV/vaje/vaja4/hotel.jpg>

Na tej sliki boste poiskali robove. To boste storili na dva načina. Najprej boste uporabili Sobelov operator, o katerem si lahko več preberete v Wikipediji:

http://en.wikipedia.org/wiki/Sobel_filter

Na tej strani boste našli dokumentirano tudi jedro Sobelovega filtra, ki ga na sliko aplicirate s pomočjo matlabovih funkcij `conv2` ali `imfilter`. Ne uporabljajte Matlabove funkcije `fspecial` - jedro filtra določite sami na podlagi tistega, kar preberete na Wikipediji. Ko ste tako dobili magnitudo gradientov G , jo obravnajte kot sivinsko sliko, izberite ustrezen prag in izvedite upravljanje. Rezultat naj je binarna slika, ki ima vrednosti 1 tam, kjer se nahajajo robovi, drugje pa naj je vrednost slikovnih elementov 0.

Drug način, ki ga morate izvesti, je z uporabo Cannyevega detektorja robov. O njem si lahko več preberete tule:

http://en.wikipedia.org/wiki/Canny_edge_detector

Uporabili boste različico detektorja, ki je že vgrajena v Matlabov Image processing toolbox. To storite z uporabo funkcije `edge`, vendar pazite, da boste izbrali Cannyev detektor. Funkcija vam že da binarno sliko. Sliki prikazite eno ob drugi in ju primerjajte.

2 Houghova transformacija (40%)

Na obeh slikah robov iz prejšnje točke boste izvedli Houghovo transformacijo za iskanje premic. Houghovo transformacijo morate sprogramirati sami – v tej točki ne smete uporabiti vgrajene Matlabove funkcije, ali zunanje kode, ki ste jo dobili na Internetu. Kodo, ki jo boste napisali, boste namreč morali tudi podrobno razumeti in na zagovoru to pokazati. Nekaj več o Houghovi transformaciji najdete tule:

http://en.wikipedia.org/wiki/Hough_transform

Nalogo rešite na naslednji način. Predpostavite naslednji model premice: $y = kx + n$. Vaša parametra Houghovega prostora sta torej k in n . Parameter k iščite v območju od -2 do 2 (interval razdelite na 400 delov), parameter n pa od -200 do 200 (tudi ta interval razdelite na 400 delov). Če se izkaže, da je potrebno območje ali ločljivost akumulatorja spremeniti za dobre rezultate, to storite. Sprogramirajte funkcijo, ki s pomočjo rezultatov Houghove transformacije najde N najbolj izrazitih premic ter jih izriše čez originalno sliko. Ta funkcija naj kot parametre vzame originalno sliko, rezultat Houghove transformacije in N . Delovanje preizkusite na obeh slikah robov iz prejšnje naloge.

3 Detekcija področij in morfološke operacije (40%)

Pri tej vaji boste uporabljali posnetek tekme squash, ki ga snamete iz naslednjega naslova:

<http://vision.fe.uni-lj.si/classes/RV/vaje/vaja4/squash.avi>

Posnetek predvajajte in se spoznajte z naravo prizora. Na sliki sta dva igralca na modrem ozadju, en nosi belo majico, drug pa rdečo. Med igro sta večinoma oddaljena en od drugega, nekajkrat pa pride med njima do kontakta.

Vaša naloga je narediti enostaven detektor igralcev, s katerim boste lahko dobili trajektorije gibanja igralcev (zaporedje lokacij (x, y)) v slikovnih koordinatah. To boste naredili na naslednji način.

Iz posnetka morate najprej izločiti sliko praznega igrišča. Kot lahko vidite, igrišče na posnetku nikoli ni prazno. Uporabili boste znanje o prizoru – igralca na sliki se ves čas premikata, zato je verjetno, da bi statistika vsakega slikovnega elementa, ki bi ga opazovali skozi čas, pokazala, da je slikovni element veliko večino časa ene same barve – barve ozadja. Spomnite se predavanj prof. Kovačiča in izračunajte sliko ozadja s pomočjo enega samega klica ene od statističnih funkcij, ki so že vgrajene v Matlab (pozor, sliko ozadja boste morali še pretvoriti v 8 bitni celoštevilski format s pomočjo funkcije `uint8`, da jo boste lahko prikazali). Za izračun ozadja ne uporabite celega posnetka, ampak izberite samo vsako deseto ali dvajseto sliko, torej 80 ali 40 slik, ki jih zložite v 4-D matriko. Na tej matriki uporabite izbrano statistično funkcijo.

Nato boste izvedli detekcijo igralcev s pomočjo naslednjih korakov. Najprej izračunajte sliko razlike, ki jo dobite kot evklidsko razdaljo posameznega RGB slikovnega elementa slike od RGB slikovnega elementa slike ozadja. To sliko prikažite s pomočjo funkcij `imagesc` in `colorbar`. Paleta za prikaz lahko nastavite na sivinske vrednosti s pomočjo klica `colormap gray`. Oglejte si razpon vrednosti slike razlike in določite približen prag za upravljanje, ki vam bo na binarni sliki ohranil

oba igralca, in odstranil ozadje (za ta korak je morda boljše, če ne uporabite `colormap gray`, čeprav bo slika izgledala nenaravno).

Nato na binarni sliki uporabite morfološke operacije po želji, vaš cilj pa naj je, da izboljšate binarno sliko tako, da bo na njej čimmanj šuma, po drugi strani pa naj sta igralca čimbolj vidna. Za to uporabite Matlabovo funkcijo `bwmorph` z ustrežno izbiro parametrov. Na koncu s pomočjo funkcije `bwlabel` najдите poziciji (ali očrtana pravokotnika ali težišči) največji zaključenih področij na sliki – če ste vse naredili prav, bosta to že poziciji igralcev v slikovnih koordinatah. Te pozicije vrišite na izvorno sliko. Tako obdelajte celoten posnetek in naredite novo `.avi` datoteko, kjer bodo čez igralca narisane njune pozicije.

4 Dodatna naloga (20%)

Igralcev na posnetku iz prejšnje naloge ne morete identificirati samo s pomočjo slike razlike. Zato s pomočjo znanja, pridobljenega na prejšnji Vaji 3 metodo nadgradite tako, da bo na podlagi barve majice, ki jo nosi igralec, določila njegovo identiteto. Lahko uporabite poljubno metodo – ali ročno določanje mej v RGB prostoru, ali pa barvni histogram. Posnetek, ki ga naredite s tako popravljeno metodo, naj ima eksplicitno označeno tudi identiteto igralca (recimo 1 in 2, ali A in B).