

Távérzékelés második beadandó

Készítette: Sebestyén Gábor (segabor@elte.hu)
Honlap: <http://segabor.web.elte.hu/taverzekeles/>

Alkalmazott módszerek, technikák és algoritmusok

1. ImageMagick programcsomag (<http://www.imagemagick.org>)
A program alapvető képmanipulálásokra az ImageMagick programcsomag API-ját használja. Ennek segítségével olvassa be a képet, transzformálja és menti ki.
2. Pont teszt poligonban: PNPOLY - Point Inclusion in Polygon Test, W. Randolph Franklin (http://www.ecse.rpi.edu/Homepages/wrf/Research/Short_Notes/pnpoly.html)
3. χ^2 (khi-négyszet) függvény táblák (<http://www.ento.vt.edu/~sharov/PopEcol/tables/chisq.html>)

Implementáció

Az implementáció ruby szkript nyelven készült, erősen támaszkodva az ImageMagick programcsomag szolgáltatásaira.

Megvalósított osztályok és feladataik

1. request.rb - kéréseket tartalmazó osztály
2. plant.rb - növény fajtáját megvalósító osztály (statisztikák, átlagvektor, kód, stb.)
3. chi_square.rb - khi-négyszet táblázatok és lineáris kereső
4. polygon.rb - poligon osztály
5. vertex.rb - csúcspont osztály
6. util.rb - egyéb segédeljárások és bővítések

Telepítés

A program külön telepítés nem szükséges (a támogató csomagok kivételével). Szükséges könyvtárak és programcsomagok:

1. ImageMagick - programcsomag
2. ruby 1.8.x - szkript nyelv
3. rubygems - ruby csomagkezelő program
- 3.1 rmagick - ImageMagick ruby nyelvű kiegészítése
telepítése rubygems-en keresztül javasolt (`gem install rmagick`)

Futtatás

A program a saját "bead1" mappájából futtatható parancssorból.

```
ruby bead2.rb
```

Ha paramétert nem adunk meg, a program a bead2.rb-t tartalmazó mappában a *request.tif* alatti mappában keresi az input fájlokat (TIFF-ek és request.txt):

```
./request.tif:  
im0317.tif im0606.tif im0727.tif reference.tif  
im0506.tif im0630.tif im0811.tif request.txt
```

A futás során az egyes lépéseket [i] ... formában írja ki, ahol i a lépésszámot jelenti.

A lefutás végén az indexelt kimenet képet megkapjuk .gif és .tif formátumú fájlokban, továbbá result.txt-ben a kimentett statisztikát.

Egy futtatás kimenete:

```
[1] loading tif files  
[1.1] Image size: 841 x 921  
[2] processing requests from 'request.tif/request.txt'  
Number of requests: 12  
[2.x.1] 13 => color: magenta  
[2.x.1] 1 => color: yellow  
[2.x.1] 3 => color: blue  
[2.x.1] 1 => color: yellow  
[2.x.1] 15 => color: pink  
[2.x.1] 1 => color: yellow  
[2.x.1] 14 => color: green  
[2.x.1] 14 => color: green  
[2.x.1] 10 => color: red  
[2.x.1] 12 => color: sienna  
[2.x.1] 13 => color: magenta  
[2.x.1] 40 => color: green  
[3] creating output image  
[4] sampling reference image by plant identifiers  
.. 0% done in 0.125679 secs  
.. 2% done in 2.596139 secs  
.. 4% done in 2.283462 secs  
.. 6% done in 2.074729 secs
```

```
.. 8% done in 1.90521 secs
.. 10% done in 2.053616 secs
.. 13% done in 1.816803 secs
.. 15% done in 2.006005 secs
.. 17% done in 2.379794 secs
.. 19% done in 2.780136 secs
.. 21% done in 3.065056 secs
.. 23% done in 2.94181 secs
.. 26% done in 2.893258 secs
.. 28% done in 3.164784 secs
.. 30% done in 2.276241 secs
.. 32% done in 1.791578 secs
.. 34% done in 2.209684 secs
.. 36% done in 2.403069 secs
.. 39% done in 2.294121 secs
.. 41% done in 2.532266 secs
.. 43% done in 2.495279 secs
.. 45% done in 2.498504 secs
.. 47% done in 2.338252 secs
.. 49% done in 2.430312 secs
.. 52% done in 1.872871 secs
.. 54% done in 1.600673 secs
.. 56% done in 1.480117 secs
.. 58% done in 2.124483 secs
.. 60% done in 2.005379 secs
.. 62% done in 1.91428 secs
.. 65% done in 1.986578 secs
.. 67% done in 1.747337 secs
.. 69% done in 2.036391 secs
.. 71% done in 1.929786 secs
.. 73% done in 1.695614 secs
.. 76% done in 1.680736 secs
.. 78% done in 1.858118 secs
.. 80% done in 1.919385 secs
.. 82% done in 2.20184 secs
.. 84% done in 2.180309 secs
.. 86% done in 2.454097 secs
.. 89% done in 2.308527 secs
.. 91% done in 2.366316 secs
.. 93% done in 2.770387 secs
.. 95% done in 2.514627 secs
.. 97% done in 3.290927 secs
.. 99% done in 3.466317 secs
[4.1] process done in 105 seconds
[4.2] plant magenta has 11620 samples
[4.2] plant yellow has 27090 samples
[4.2] plant blue has 10064 samples
[4.2] plant pink has 448 samples
[4.2] plant green has 7309 samples
[4.2] plant red has 10105 samples
[4.2] plant sienna has 9900 samples
[4.2] plant green has 0 samples
[4.3] colors not used: ["DarkGreen", "orange", "aquamarine", "gray69"]
[5] calculating covariance matrices for each plant
[5.1] plant 13
.. done in 34 secs
[5.1] plant 1
.. done in 81 secs
[5.1] plant 3
.. done in 30 secs
[5.1] plant 15
.. done in 1 secs
[5.1] plant 14
.. done in 23 secs
[5.1] plant 10
.. done in 32 secs
[5.1] plant 12
.. done in 32 secs
[6] processing requests
[6.1] processing request 1001 ..
.. done in 13 secs
[6.1] processing request 1002 ..
.. done in 14 secs
[6.1] processing request 1003 ..
.. done in 4 secs
[6.1] processing request 1004 ..
.. done in 5 secs
[6.1] processing request 1005 ..
.. done in 3 secs
[6.1] processing request 1006 ..
.. done in 3 secs
[6.1] processing request 1007 ..
.. done in 3 secs
```

```
[6.1] processing request 1008 ..  
.. done in 3 secs  
[6.1] processing request 1009 ..  
.. done in 2 secs  
[6.1] processing request 1010 ..  
.. done in 2 secs  
[6.1] processing request 1011 ..  
.. done in 23 secs  
[7] writing results out  
... gif  
... tif  
1001 87 0 13  
1002 99 0 0  
1003 95 0 5  
1004 94 2 4  
1005 0 0 100  
1006 80 4 16  
1007 2 2 96  
1008 33 36 31  
1009 100 0 0  
1010 99 0 1  
1011 98 0 2  
1012 0 0 0  
[8] done  
Program exited.
```