Practica 3.1

1) Implement a script that calculates either "Haar Features" and/or HOG, or any other descriptor on an image.

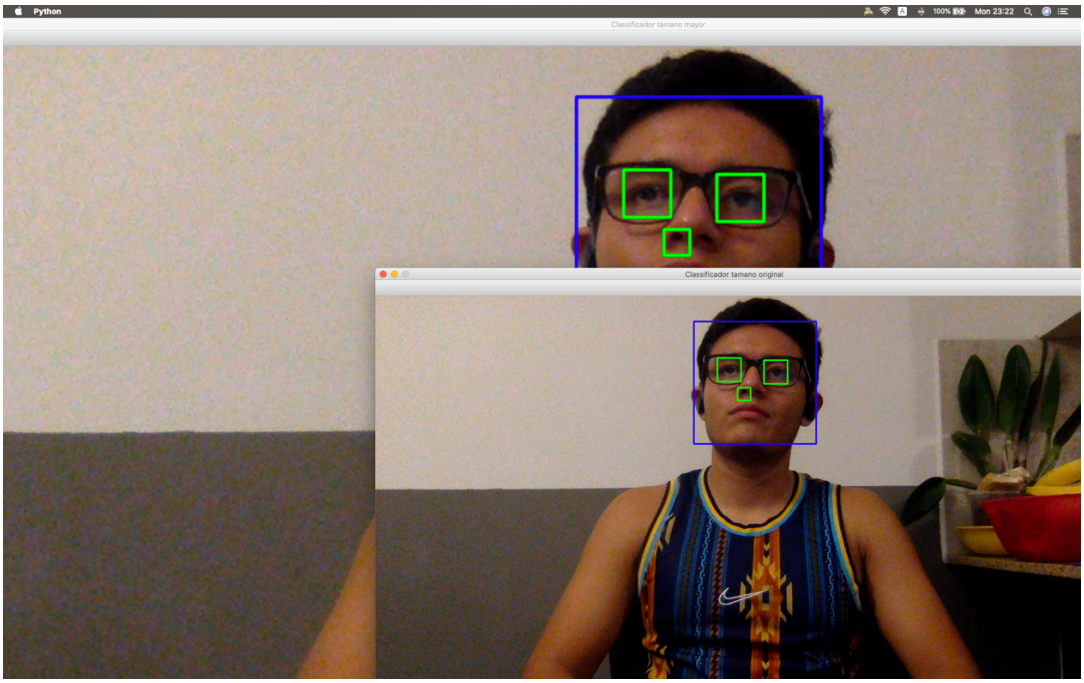
2) Then apply on the features obtained the Sliding pyramid window algorithm.

3) For every sample:
   - Save the original fragment of the image.
   - Pass from Matrix to Vector the corresponding elements of the obtained characteristics and save them in the format or data structure that you think is appropriate.

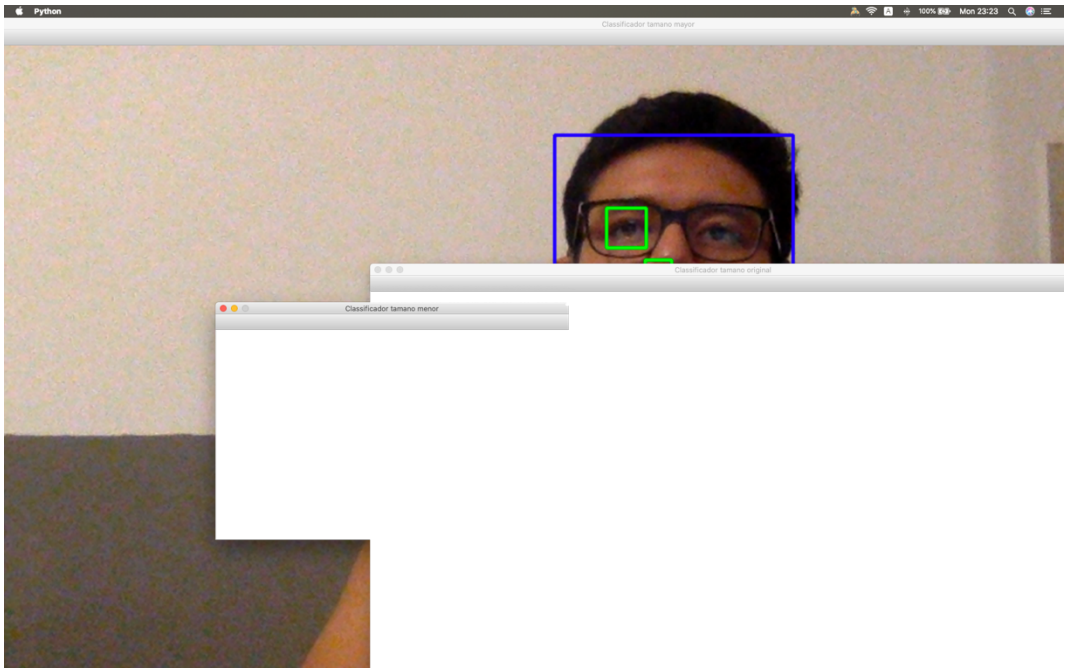
1 & 2)

Había una imagen mas pequeña pero no lo dejaba imprimir en captura de pantalla ☹

Pyramid Upper and Normal Image

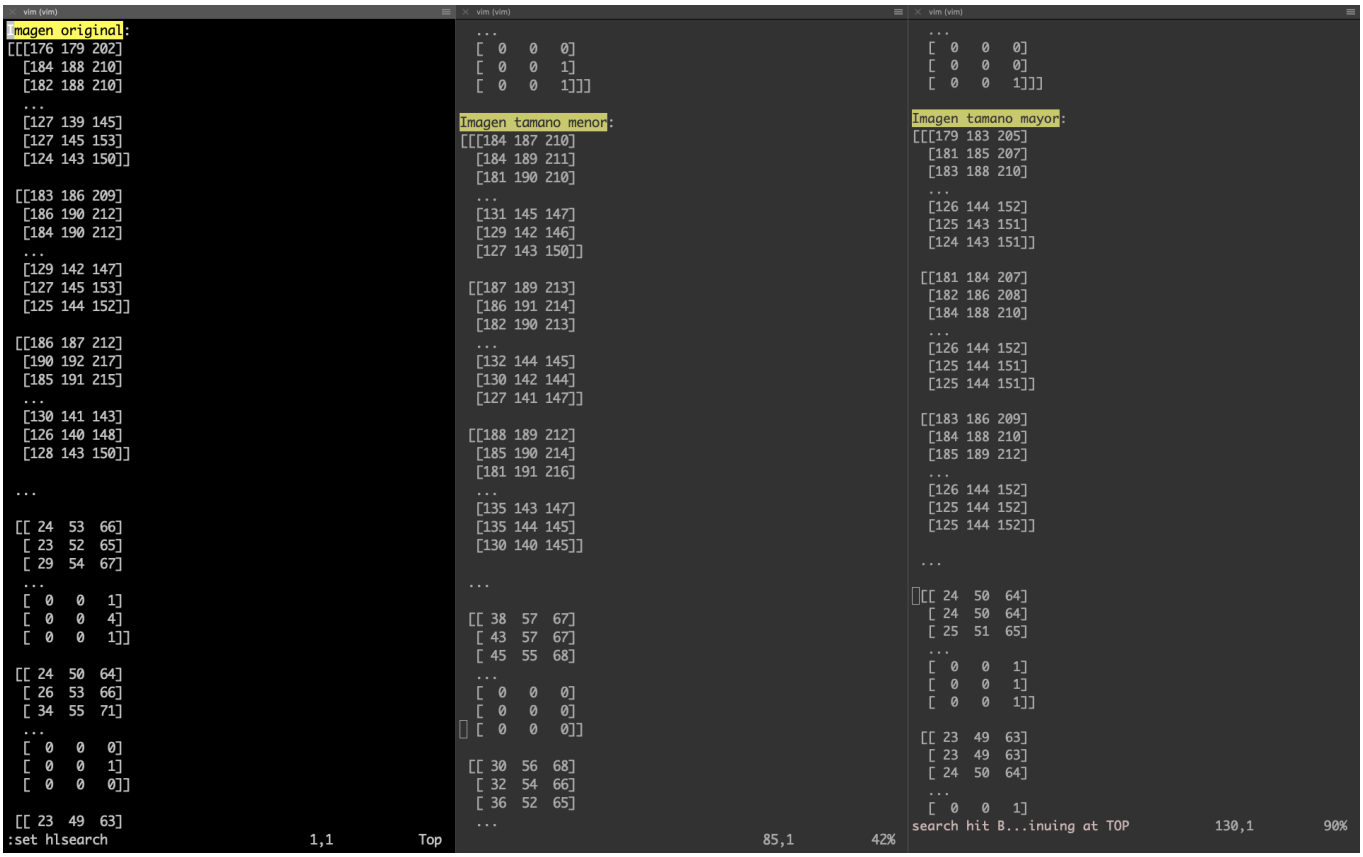


Lower, Upper and Normal images



3.) Images saved in a file as Vectors

```
➜  07 git:(master) ✗ ll
total 2608
-rw-r--r--  1 memoherrera  staff   333K Nov  2 23:04 haarcascade_eye.xml
-rw-r--r--  1 memoherrera  staff   908K Nov  2 23:04 haarcascade_frontalface_default.xml
-rw-r--r--  1 memoherrera  staff   1.9K Nov  2 23:32 imagenes.txt
-rw-r--r--  1 memoherrera  staff   1.5K Nov  2 23:32 practica3_1.py
➜  07 git:(master) ✗
```


vim (vim)
Imagen original:
[[[176 179 202]
[184 188 210]
[182 188 210]
...
[127 139 145]
[127 145 153]
[124 143 150]]
[[183 186 209]
[186 190 212]
[184 190 212]
...
[129 142 147]
[127 145 153]
[125 144 152]]
[[186 187 212]
[190 192 217]
[185 191 215]
...
[130 141 143]
[126 140 148]
[128 143 150]]
...
[[24 53 66]
[23 52 65]
[29 54 67]
...
[0 0 1]
[0 0 4]
[0 0 1]]
[[24 50 64]
[26 53 66]
[34 55 71]
...
[0 0 0]
[0 0 1]
[0 0 0]]
[[23 49 63]
:set hlsearch
1,1
Top
vim (vim)
...
[0 0 0]
[0 0 1]
[0 0 1]]]
Imagen tamano menor:
[[[184 187 210]
[184 189 211]
[181 190 210]
...
[131 145 147]
[129 142 146]
[127 143 150]]
[[187 189 213]
[186 191 214]
[182 190 213]
...
[132 144 145]
[130 142 144]
[127 141 147]]
[[188 189 212]
[185 190 214]
[181 191 216]
...
[135 143 147]
[135 144 145]
[130 140 145]]
...
[[38 57 67]
[43 57 67]
[45 55 68]
...
[0 0 0]
[0 0 0]
[0 0 0]]
[[30 56 68]
[32 54 66]
[36 52 65]
...
85,1
42%
vim (vim)
...
[0 0 0]
[0 0 0]
[0 0 1]]]
Imagen tamano mayor:
[[[179 183 205]
[181 185 207]
[183 188 210]
...
[126 144 152]
[125 143 151]
[124 143 151]]
[[181 184 207]
[182 186 208]
[184 188 210]
...
[126 144 152]
[125 144 151]
[125 144 151]]
[[183 186 209]
[184 188 210]
[185 189 212]
...
[126 144 152]
[125 144 152]
[125 144 152]]
...
[[24 50 64]
[24 50 64]
[25 51 65]
...
[0 0 1]
[0 0 1]
[0 0 1]]
[[23 49 63]
[23 49 63]
[24 50 64]
...
[0 0 1]
search hit B...inuing at TOP
130,1
90%