Laboratorio: Usos reales de la morfología matemática

**Objetivos**

Este laboratorio persigue un objetivo sencillo. Dada la siguiente imagen, contar el número de círculos que presenta la imagen.

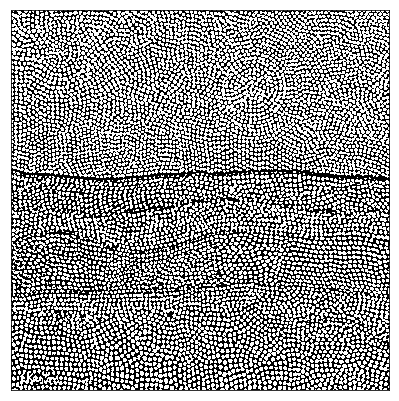


**Descripción** de la actividad

Esta imagen, presenta varios problemas de procesamiento de imagen, de los cuales, principalmente se pretenden cubrir dos en este laboratorio:

* Si buscamos umbralizar la imagen, será complicado si se utilizan métodos no adaptativos, ya que existen diferentes colores dentro de la imagen. Para ello, se deberá hacer uso de técnicas de umbralización que eviten este efecto y devuelvan una imagen parecida a la siguiente.

De esta manera, habremos resuelto el problema de los diferentes colores, y ahora sólo nos quedará contar los círculos.



* Muchos de estos círculos están colindantes entre sí, y algunos se tocan, luego, la estrategia de contar directamente el número de objetos dentro de la imagen no parece ser la mejor solución (no sería la más precisa). Por ello, se propone hacer uso de técnicas de morfología matemática para poder reducir esa parte colindante, y poder contar los objetos por separado.

Se presenta un notebook con todos los paquetes necesarios y funciones ya creadas para facilitar al alumno/a el comienzo.

**Entrega**

El entregable consistirá en una memoria, en PDF, donde se analice el código escrito y las salidas tanto de la imagen umbralizada como del número de círculos contados, así como una descripción detallada de cómo se ha resuelto.

El alumno/a deberá probar varias técnicas y detallar con cuál ha obtenido mejores resultados.

En caso de que no se detalle la descripción de cómo se ha resuelto, no se obtendrá la máxima nota.

Rúbrica

|  |  |  |  |
| --- | --- | --- | --- |
| Actividad Grupal  (valor real: 4  puntos) | Descripción | Puntuación máxima  (puntos) | Peso  % |
| Criterio 1 | Inicio del ejercicio con carga de la imagen, importación de paquetes, etc… | 1 | 10% |
| Criterio 2 | Empleo de algoritmos de umbralización adaptativa basada en color | 3 | 30% |
| Criterio 3 | Empleo de morfología matemática para maximizar el número de puntos encontrados | 4 | 40% |
| Criterio 4 | Documentar adecuadamente el ejercicio justificando las decisiones tomadas (métodos empleados, filtros, etc…) | 2 | 20% |
|  |  | **10** | **100 %** |

<https://www.youtube.com/watch?v=1owu136z1zI>

<https://github.com/GuoleiSun/CountSeg/blob/master/demo/images/demo-pascal.ipynb>

<https://www.analyticsvidhya.com/blog/2019/02/building-crowd-counting-model-python/>

<https://github.com/cfotache/pytorch_objectdetecttrack/blob/master/PyTorch_Object_Detection.ipynb>

<https://github.com/Cyanogenoid/vqa-counting>

<https://github.com/eriklindernoren/PyTorch-YOLOv3>

<https://github.com/GuoleiSun/CountSeg/blob/master/demo/images/demo-pascal.ipynb>

Finales

\*\*

<https://mathworld.wolfram.com/CirclePacking.html>

<https://scipython.com/blog/packing-circles-in-a-circle/>

<https://het.as.utexas.edu/HET/Software/Numpy/reference/generated/numpy.random.choice.html>

<https://stackoverflow.com/questions/38848759/valueerror-all-the-input-arrays-must-have-same-number-of-dimensions>

<https://scikit-image.org/docs/stable/auto_examples/segmentation/plot_regionprops.html>

<https://en.wikipedia.org/wiki/Otsu%27s_method>

<https://scikit-image.org/docs/dev/api/skimage.filters.html#skimage.filters.threshold_local>