

Reto 5 - Meanshift y Camshift

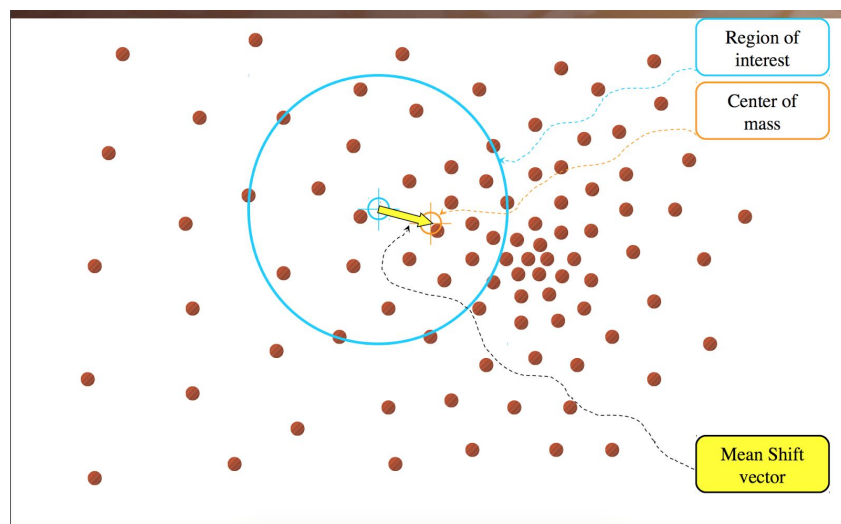
(Jueves 15 Febrero)

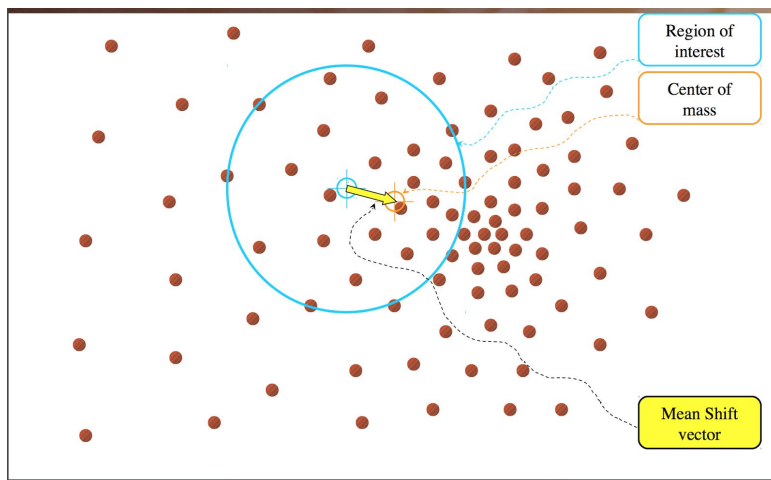
5.1. Meanshift

Método iterativo no paramétrico que puede ser usado para clustering, segmentación, etc. Encuentra las modas de unas distribuciones, pero sin necesitar saber cuántas modas tenemos. Considera que el espacio de datos es una función de densidad de probabilidad muestreada. Para cada punto del conjunto de datos, encuentra la moda más cercana. Para ello, define una región alrededor de ese punto y encuentra su media, cambiando la situación de la media actual a la nueva (shift). Repite el proceso hasta que converja.

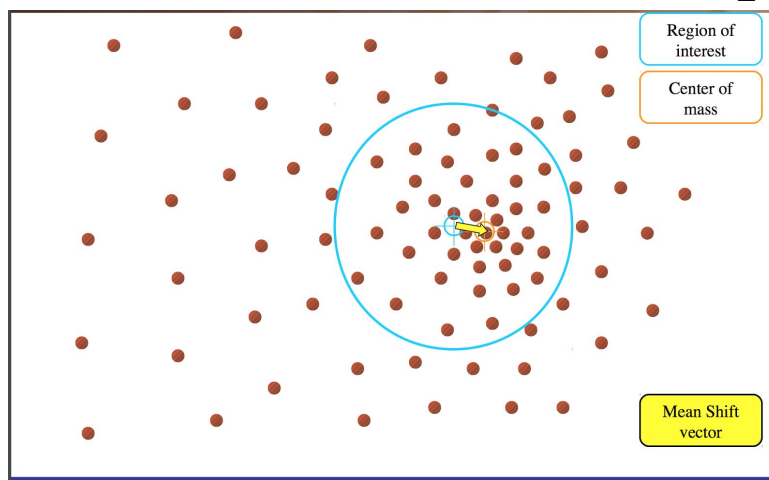
A continuación se detallan una serie de imágenes par explicar mejor este proceso:

1. Elegimos un punto desde el cual empezar y establecemos un radio que consideramos la 'Región de interés', y dentro de ese área hacemos la moda de puntos, viendo donde se sitúan la mayor densidad de puntos. En ese momento establecemos el 'centro de la masa'. El cual será el punto referencia desde donde volveremos a iterar, para conseguir acercarnos finalmente al punto con más densidad de puntos de toda la imagen.

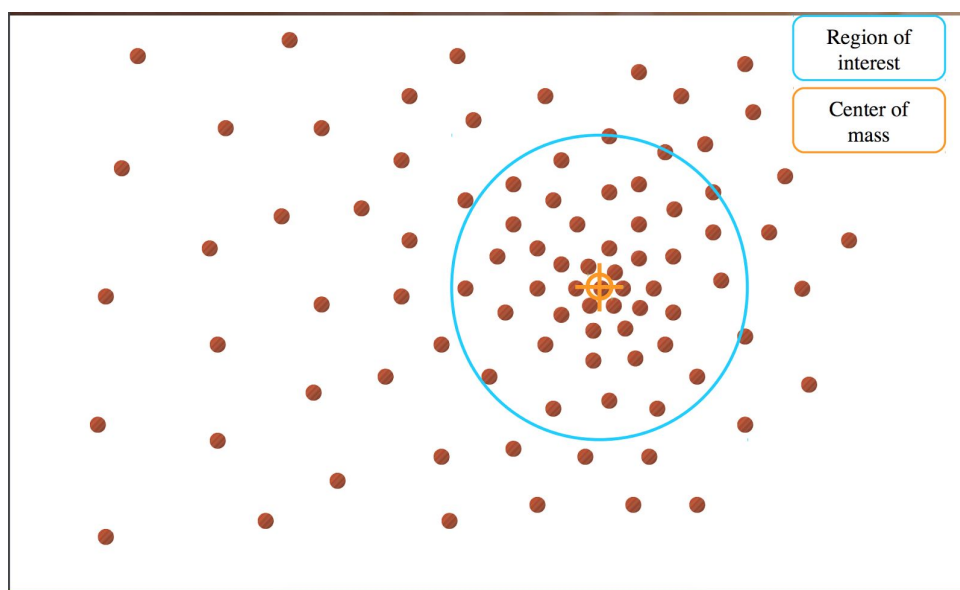




current_account_info



Dando como resultado el centro con mayor densidad de puntos.



Es un proceso a nivel teórico bastante más complejo, lo cual no entramos en más detalle teórico.

https://github.com/paradigmadigital/hackathon-image-processing/blob/master/tip_02_01_a.py

En esta práctica se setea inicialmente los valores iniciales del cuadro a detectar. Es interesante que intentéis sacar esos puntos de manera dinámica con los retos realizados anteriormente.

Bibliografía

<https://rua.ua.es/dspace/bitstream/10045/17323/6/segmentacion.pdf>

https://docs.opencv.org/3.4.0/db/df8/tutorial_py_meanshift.html

5.2. Camshift

El algoritmo de Meanshift es muy interesante, pero nuestra ventana siempre tiene el mismo tamaño cuando el objeto está más lejos o está muy cerca de la cámara. Eso no es bueno. El escenario ideal sería poder adaptar el tamaño de la ventana con el tamaño y rotación del objetivo. La solución se llama CAMshift (Continuously Adaptive Meanshift).

Aplica el cambio de significado primero. Una vez que mediashift converge, actualiza el tamaño de la ventana como, $s = 2 \times \sqrt{M_{00} / 256}$. También calcula la orientación de la elipse que mejor se adapte a ella. De nuevo se aplica la mediashift con la nueva ventana de búsqueda a escala y la ubicación de la ventana anterior. El proceso continúa hasta que se alcanza la precisión requerida.

https://github.com/paradigmadigital/hackathon-image-processing/blob/master/tip_02_01_b.py
https://www.youtube.com/watch?time_continue=3&v=6rcTz1luPGw