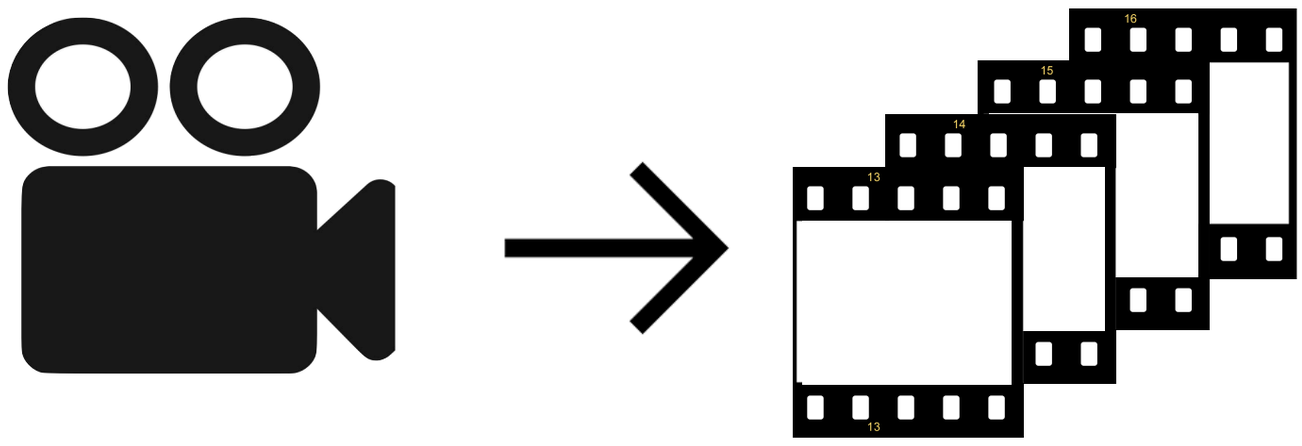
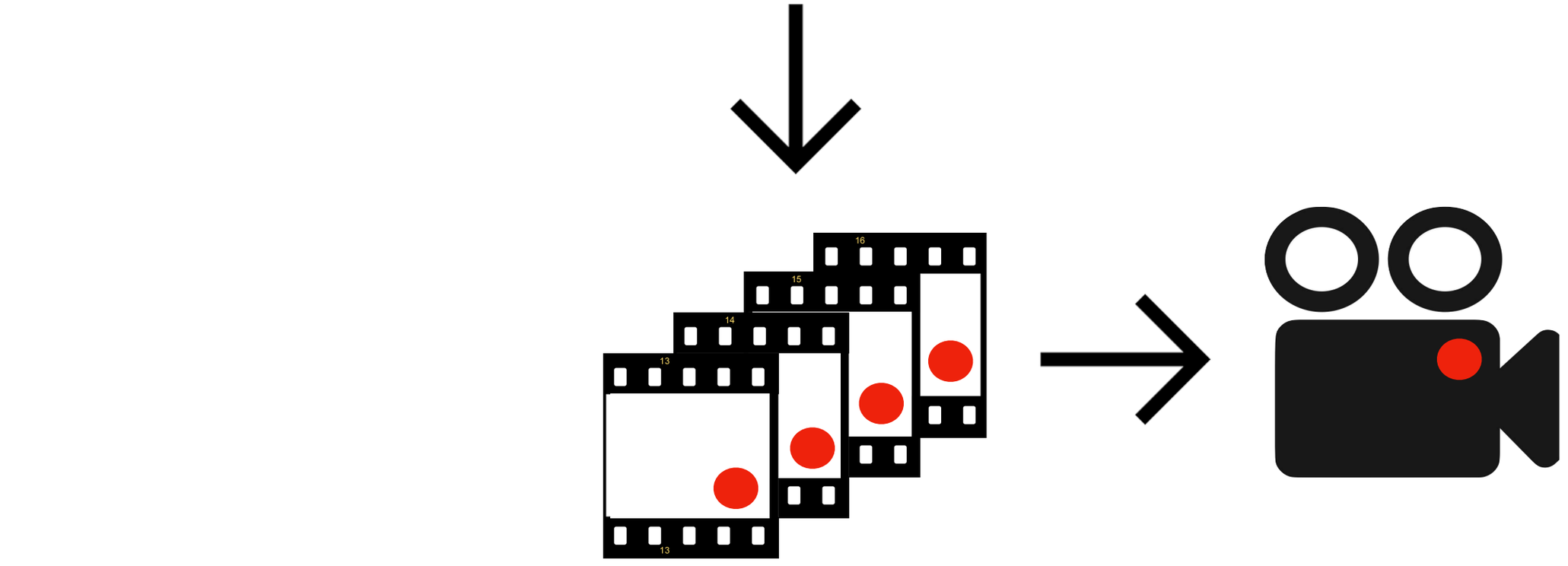
A high performance video segmentation framework

Liudmila Karagyaur
Lorenzo Ferri
Vanessa Braglia

Idea





Clustering

with k-means

Iteratively minimise the distance between each point and the **centroids**

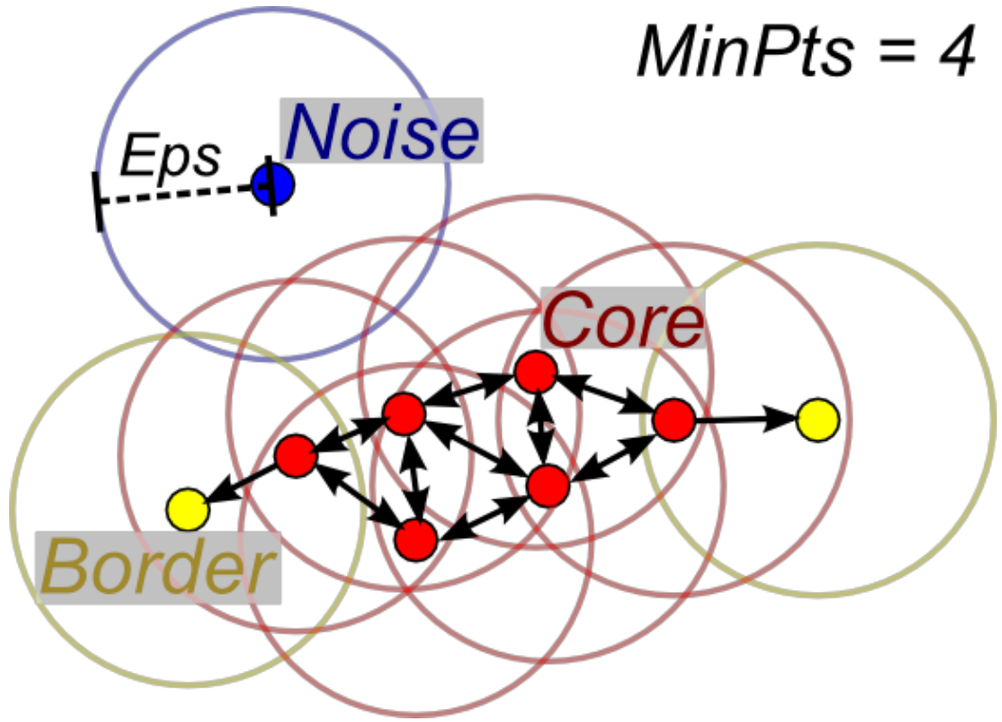
$$\arg\min_j D(x_i, c_j) \quad j = 1, \ldots, k$$

and compute the new **centroids** by the mean of the clusters

$$c_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_i$$

Clustering

with DBSCAN



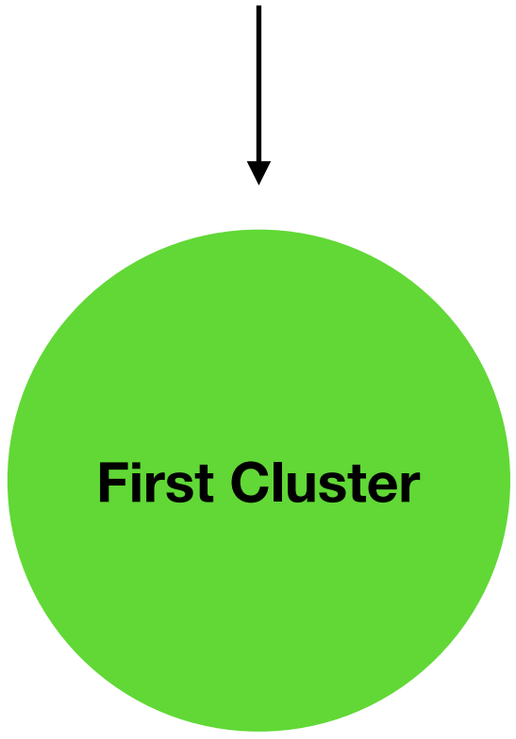
The **core** points together with the **border** points will form the clusters. The **noise** points will be discarded.

Clustering

with Spectral partitioning

This algorithm need the laplacian matrix of the image.
The second smallest eigenvector **x2** is then used to create the clusters.

If x2(p) < 0



If x2(p) > 0

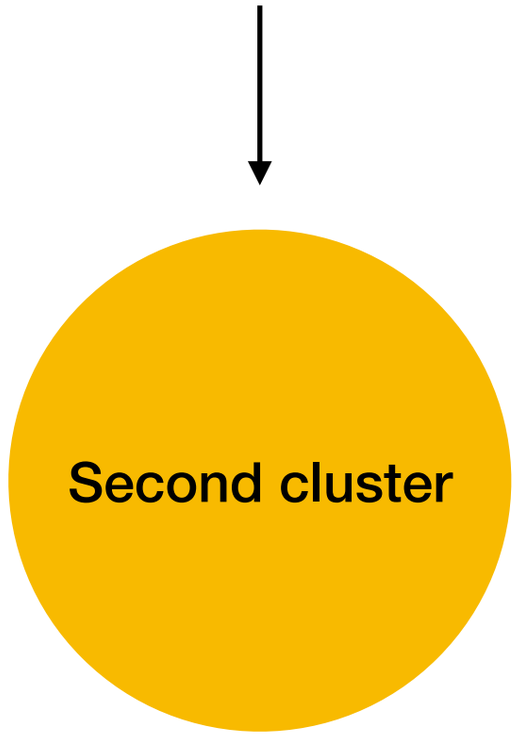


Image Sharpening

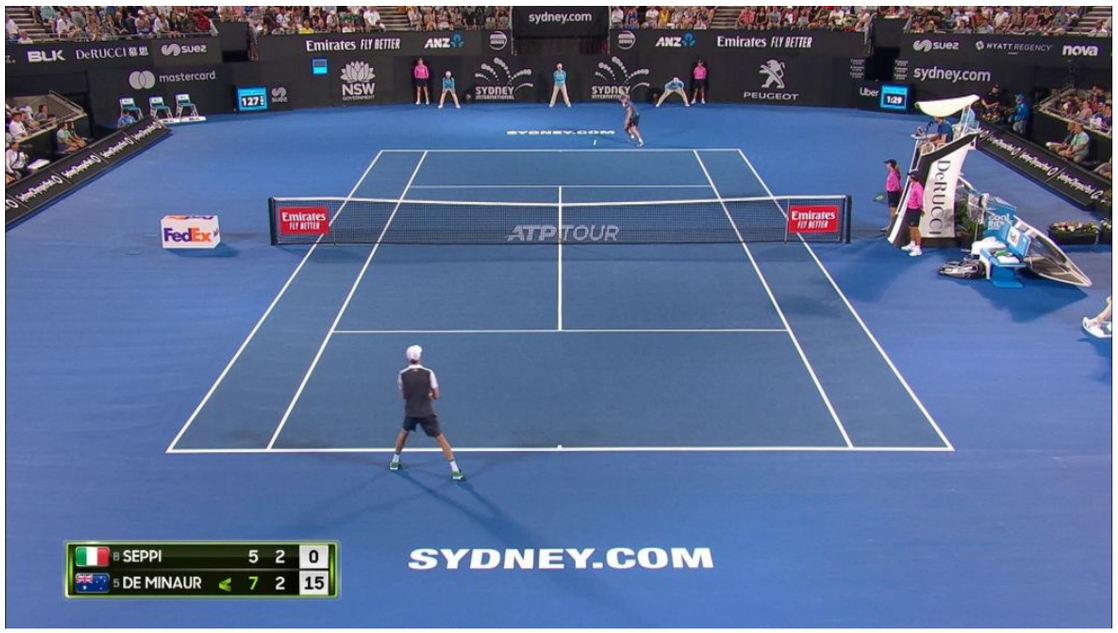
A ?

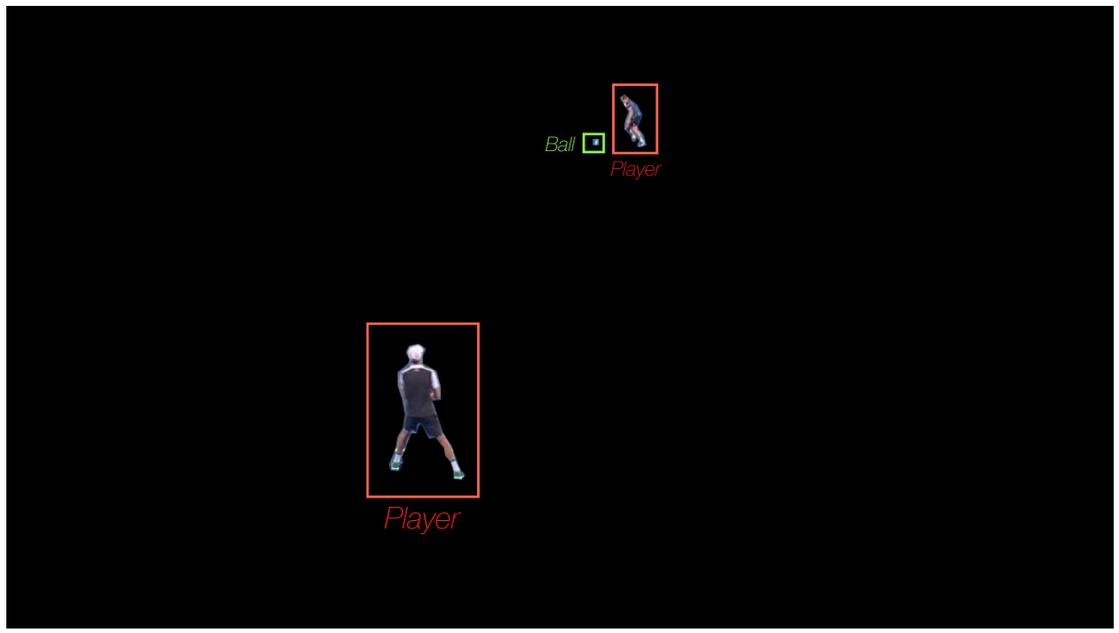
b

x

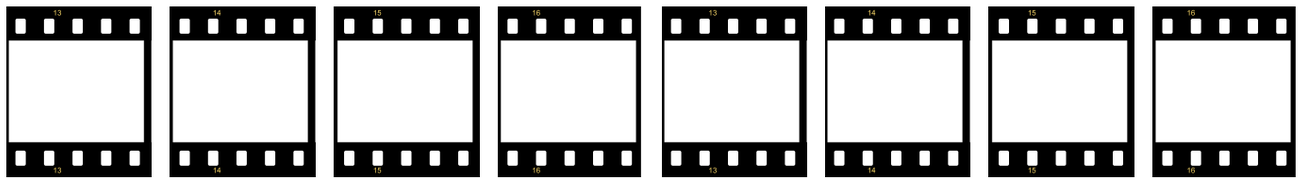
A x = b

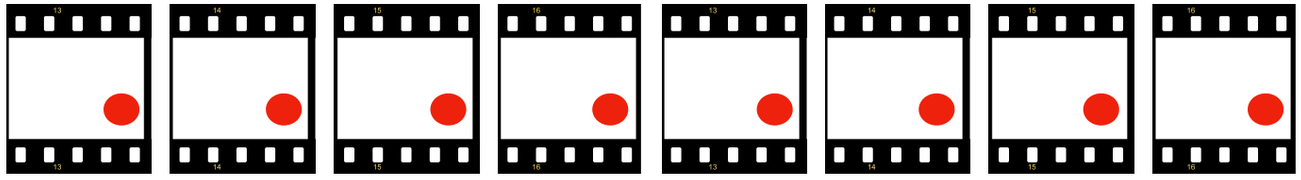
Machine Learning

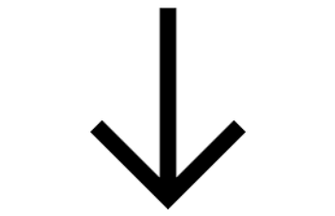


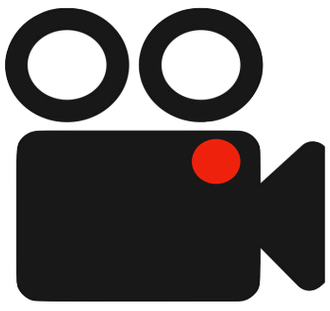


Parallelising









Project Time Line

Project TImeLine

	W1 18.02 — 22.02	W2 25.02 — 1.03	W3 4.03 — 8.03	W4 11.03 —15.03	W5 18.03 — 22.03	W6 25.03 — 29.03	W7 1.04 — 5.04	W8 8.04— 12.04	W9 15.04 —19.04	W10 21.04 — 26.04	W11 29.04 — 3.05	W12 6.05 — 10.05	W13 13.05 — 17.05	W14 20.05 — 24.05	W15 27.05— 29.05
Project setup	X	X								Holydays					
Project Plan			X	X											
From video to frame															
Clustering															
Algorithms															
Convergence analysis															
Sharpening															
Operators															
Analysis															
Machine Learning															
Parallelization															
Video Assembly															
Report															
Final Presentation															