

ESERCITAZIONE:

Ricerca del Best Match di un'Immagine Target con K-Means

Obiettivo:

Scrivere un programma in C++ che utilizzi l'algoritmo K-Means e la libreria OpenCV, per individuare l'immagine più simile a un'immagine originale (*target*) all'interno di un database di immagini.

Viene fornito il dataset (cartella *data*) che contiene l'immagine target ed una serie di altre immagini con le quali si vuole eseguire il confronto, individuando il *best match*.

Viene inoltre fornito un sorgente base (*simple.cpp*) da cui partire ed il *CMakeLists.txt*.

Approccio generale da adottare:

Fase 1: Calcolo dei Cluster dell'Immagine Target con K-Means:

1. Caricare l'immagine target: `cv::imread()`
2. Opzionale: definire una `cv::Mat` che conterrà i pixel *clusterizzati* dell'immagine target.
3. Applicare l'algoritmo K-Means (scegliendo arbitrariamente il numero di cluster K, il valore di *attempts* e un criterio di terminazione).

```
double cv::kmeans ( InputArray      data,  
                    int              K,  
                    InputOutputArray bestLabels,  
                    TermCriteria    criteria,  
                    int              attempts,  
                    int              flags,  
                    OutputArray     centers = noArray()  
                    )
```

4. Come **criterio di terminazione** è possibile utilizzare:

```
cv::TermCriteria(cv::TermCriteria::EPS|cv::TermCriteria::MAX_ITER, 10000, 0.0001)
```

Ovvero l'algoritmo terminerà una volta che la variazione dei centri dei cluster sarà inferiore a 0.0001 (**EPS**), oppure impostando un numero massimo di iterazioni (**MAX_ITER**).

5. Suggerimento: creare una struttura per memorizzare le informazioni sui *cluster* ottenuti, ad esempio i centri di massa ed il numero di pixel.

Fase 2: Calcolo dei Cluster delle altre Immagini nel Dataset:

1. Caricare le altre immagini nel database.
2. Applicare l'algoritmo K-Means a ciascuna immagine del database (scegliendo un numero di cluster, *attempts* e criterio di terminazione arbitrario).

Fase 3: Confronto tra Cluster e Output

1. Calcolare una metrica di confronto tra i cluster delle diverse immagini (ad esempio, ordinando prima i cluster in base al numero di pixel e poi confrontando i centri di massa).
2. Identificare l'immagine nel database con la metrica di confronto avente errore minimo, indicando che è la più simile all'immagine originale.