

MMS Image Retrieval

使用

主要操作可由 `image_retrieval_gui.py` 開啟操作圖形化的介面，選擇圖片和模式後按下搜尋即可得出相似圖片的結果。

暫存檔案

所有的暫存檔案可由 `image_retrieval.py` 執行後得到，主要包括：

- `color_histogram.csv`
- `color_layout.csv`
- `dataset_sift\ukbench00000.sift~ukbench01005.sift`
- `kmean_center.csv`
- `sift.csv`

- stop_word_list.csv
- sift_with_stop_word.csv

image_retrieval_q1.py

- color_histogram

主要會計算出 `dataset` 中所有圖片的 color histogram 並輸出成 `color_histogram.csv` 的檔案。

- euclidean

計算兩張圖片的 array 距離。

- Q1_ColorHistogram

算出查詢圖片與所有圖片的距離，最後回傳距離最近的10筆。

image_retrieval_q2.py

- colorlayout

算出圖片的 color layout 並傳回 (zzY,zzCb,zzCr)。

- color_layout

產生出 `color_layout.csv`。

- *Q2ColorLayout*

算出查詢圖片與所有圖片的距離，最後回傳距離最近的10筆。

image_retrieval_q3.py

- *img_to_sift*

先將 `::dataset::` 中的所有圖片轉成sift檔存在 `::dataset_sift::` 中。

- *sift_to_arr*

抓出所有 `sift` 檔中的特徵值，並將它們全部變成一的一維的 `list`。

- *arr_to_kmeans*

將轉出的 `list` 進行分群得出分群結果，並且順便產生出 `kmean_center.csv` 的檔案。

- *kCenter_to_kmeans*

利用 `kmeans` 的中心點產生出 `kmeans` 加快處理速度。

- *img_to_vector*

將每張圖片利用 `kmeans` 產生出 128 維的向量，並產生出 `sift.csv` 檔案。

- *vector_distance*

計算向量之間的相似度。

- Q3_SIFT_Visual_Words

算出查詢圖片與所有圖片的距離，最後回傳距離最近的10筆。

image_retrieval_q4.py

- create_stop_word_list

利用128維的向量分群計算出前10%重複次數最高的群作為 stop word list，並且產生出stop_word_list.csv 檔案。

- get_stop_word_list

取的 stop_word_list.csv 檔案，回傳一個 list。

- create_vector_stop_word_list

利用 stop_word_list，重新產生出一個 sift_with_stop_word_list.csv 檔案。

- vector_distance

計算向量之間的相似度。

- Q4_Visual_Words_using_stop_words

算出查詢圖片與所有圖片的距離，最後回傳距離最近的10筆。