

---

## COMPUTER VISION - LAB 5

---

P. Zanuttigh, S. Ghidoni, U. Michieli, G. Agresti, K. Koide, M. Terreran

Topics: Keypoints, Descriptors and Matching

Goal: Create a panoramic image given a sequence of unstitched images.

Write a C++ class ( `PanoramicImage` ) which includes methods that / for:

1. Load a set of images
1. Project the images on a cylinder surface using the provided static method `cylindricalProj()` of the `PanoramicUtils` class. The method requires as a parameter an angle value (in degrees) which is half of the FoV of the camera used to take the photos. The FoV of the camera is  $66^\circ$  (half FoV= $33^\circ$ ) for all the provided datasets excluding the “dolomites” one for which it is  $54^\circ$  (half FoV= $27^\circ$ ).
2. Extract ORB or SIFT features from the images (SIFT features requires the `xfeatures2d` module that is part of the *contrib* package of OpenCV available only with installation from sources).
2. For each (consecutive) couples of images
  - a. Compute the match between the different features extracted in (3). For this, OpenCV offers you the `cv::BFMatcher` class. Remember to use L2 distance for SIFT and the Hamming distance for ORB.
  - b. Refine the matches found above by selecting the matches with distance less than  $ratio * min\_distance$ , where *ratio* is a user-defined threshold and *min\_distance* is the minimum distance found among the matches.
  - c. By the fact that in the cylinder the images are linked together by a simple translation, using the refined matches, find the translation between the images. To this end, you can use the RANSAC algorithm. While OpenCV does not provide a direct RANSAC function, the set of inliers can be computed by using the `findHomography()` function, with `CV_RANSAC` as the third parameter (hint: the inliers can be retrieved by using the mask argument). Otherwise, you can implement a simplified RANSAC following the trace on the slide.
3. Using the set of translations found in (4.c). compute the final panorama merging together the input images.

Write a program to test the previous class. The program should:

1. Create an instance of the `PanoramicImage` class with the images in the data folder
2. Display the result

SAMPLE OUTPUT:

