Computer vision course

Proff. Stefano Ghidoni | Matteo Terreran

Lab 7 - Image stitching

Task 1
Write a C++ class for stitching a set of images - i.e., creating a panorama by merging together a set of images.

To do so, you need to perform the following operations:

- Load the set of images - the images are already ordered;
- Project the images on a cylindrical surface using the cylindricalProj() function of the PanoramicUtils class. The function 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° (half FoV=33°) for all the provided datasets excluding the "dolomites" one for which it is 54° (half FoV=27°);
- Extract the SIFT features of the images (depending on the OpenCV version you are using, you may need the xfeatures2d module that is part of the contrib package of OpenCV, available only with installation from sources). You can use other features (e.g., ORB) if SIFT is not available.
- For each couple of consecutive images:
    - Compute the match between the different features extracted in the previous step. To do so, OpenCV offers you the cv::BFMatcher class (SIFT requires the use of the L2 distance).
    - 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.
    - By the fact that in the cylinder the images are linked together by a simple translation, find the translation between the images by using the refined matches. 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
- To compute the final panorama you shall create a (large enough) empty image and merge together the input images using the set of translations calculated in the previous step.

Test your algorithm using the images in the data folder.

References
Cylindrical projection document in the lab folder.
Documentation of the findHomography() function:
https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html#ga4abc2ece9fab9398f2e560d53c8c9780