Computer vision: Homework #6

Task

Given a set of objects photos and scenes pictures made from these objects, the required tasks are:

- for every object and scene couple, find and highlight the object in the scene;
- display the result as a figure with the object and the scene pictures side by side; the scene photo should have the found object in a coloured square.

1 Procedure

For this homework every dataset has been used.
The procedure is the same for every couple of images:

- After loading a dataset, the program run every object image with every scenario;
- the keypoints and descriptors have been found using an ORB detector;
- the matches are computed using a *brute force* method;
- a very simple optimization has been implemented: after sorting the matches by distance, the most distant matches are eliminated. Different thresholds have been used to compare the results;
- to display the finale image, a simple call of cv::drawMatches is not sufficient, so as suggested in the documentation, I made used of cv::findHomography() and cv::perspectiveTransform() to then draw a green box around the object in the scene image.

2 Results

This section is divided regarding the dataset used.

2.1 Dataset 1 - Random objects

There isn't much to say for *Dataset 1*, as said in the documentation, all the objects are detected even without optimization, the results are showed in Figure 1, 2, 3.

Figure 1: *Dataset 1*, detected object

Figure 2: *Dataset 1*, no object to detect

Figure 3: *Dataset 1*, detected object even if obstructed by another object

2.2 Dataset 2 - Food and a book

In the second dataset there are tree object, *Three dimensional computer vision* is detected with or without optimization, but the *Basmati rice* and *I famosi bastoncini* are more problematic. In this case, keeping the best 10% of the matches is sufficient to have good results. In the figure 4, 5, 6 the images without optimization, in 7, 8 with optimization.

Figure 4: *Dataset 2*, *I famosi bastoncini* is not detected

Figure 5: *Dataset 2*, *Three dimensional computer vision* is detected

Figure 6: *Dataset 2*, *Basmati rice* is not detected so well

Figure 7: *Dataset 2*, detected *I famosi bastoncini* after optimization

Figure 8: *Dataset 2*, detected *Basmati rice* after optimization

2.3 Dataset 3 - Cars

In the third dataset things start to be more problematic. The ORB detector does not work very well, again, as expected. In the Figures 9, 10, 11 i reported the fails, in particular in the first one there is somehow a edge of an object not present in the scene.
Removing the 50% of the matches is a big improvement, but still the green boxes area is too big and a little edge is still drawn when the object image is absent in the scene. Probably having a well cropped object image without much background would improve the detection. Results in Figures 12, 13, 14.

Figure 9: *Dataset 3*, green edge draw even if it shouldn't

Figure 10: *Dataset 3*, failed detection with blue car

Figure 11: *Dataset 3*, failed detection with red car

Figure 12: *Dataset 3*, even after optimization a little edge is still drawn

Figure 13: *Dataset 3*, better detection of the blue image after optimization

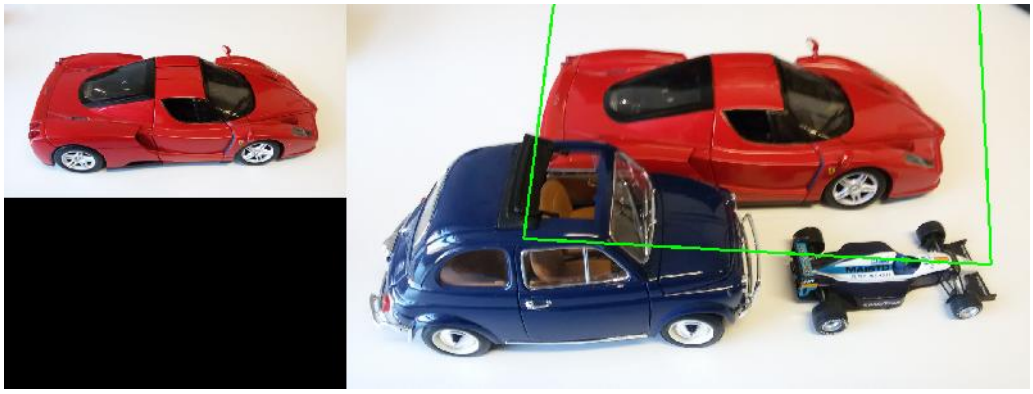

Figure 14: *Dataset 3*, better detection of the red image after optimization

2.4 Dataset 4 - Random objects

The last dataset is the most problematic, the program definitely doesn't work in a sufficient precise way so here I just report some curious results i found. Figure 15 shows how the bear with a blue papillon is detected without optimization, but in 16, removing just the 50% of the matches is enough to ruin the detection, so it means it requires more matches than the previous objects to be detected or the ORB is not good enough. Instead, The big bear with a red papillon in Figure 16 is detected if we consider only 10% of the best matches; results are displayed in Figure 18.

Figure 15: *Dataset 4*, the bear with a blue papillon is detected without an optimization

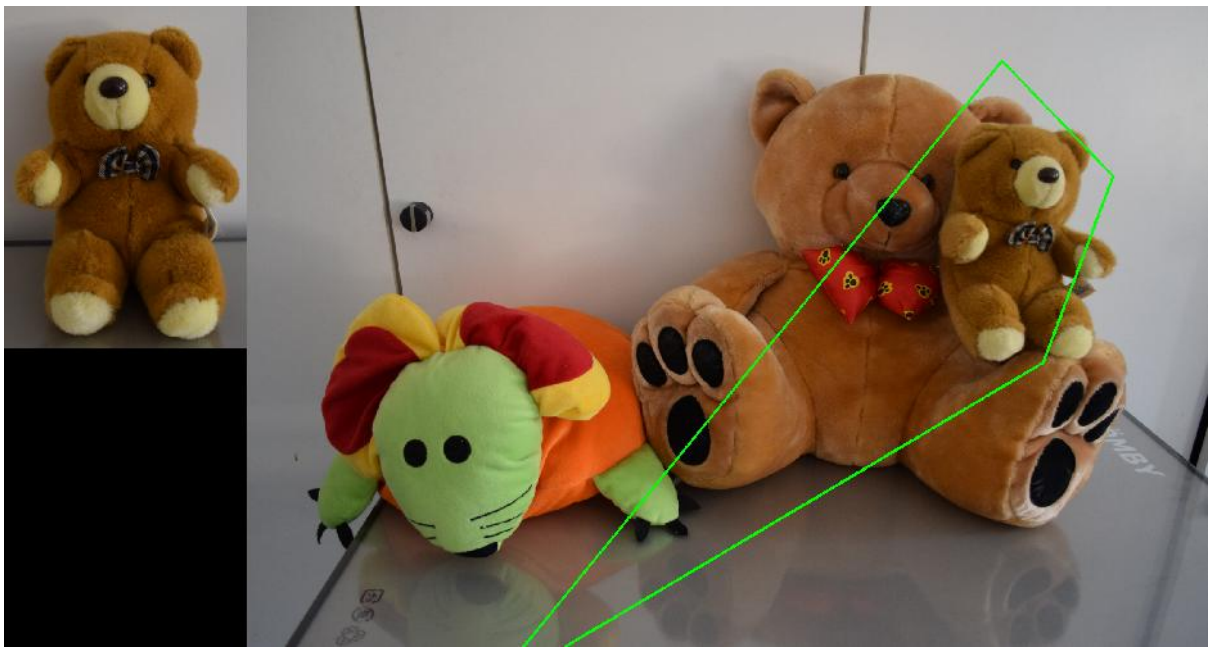

Figure 16: *Dataset 4*, the program fails to find the bear with a blue papillon removing 50% of the matches

Figure 17: *Dataset 4*, the big bear with a red papillon is not detected without optimization

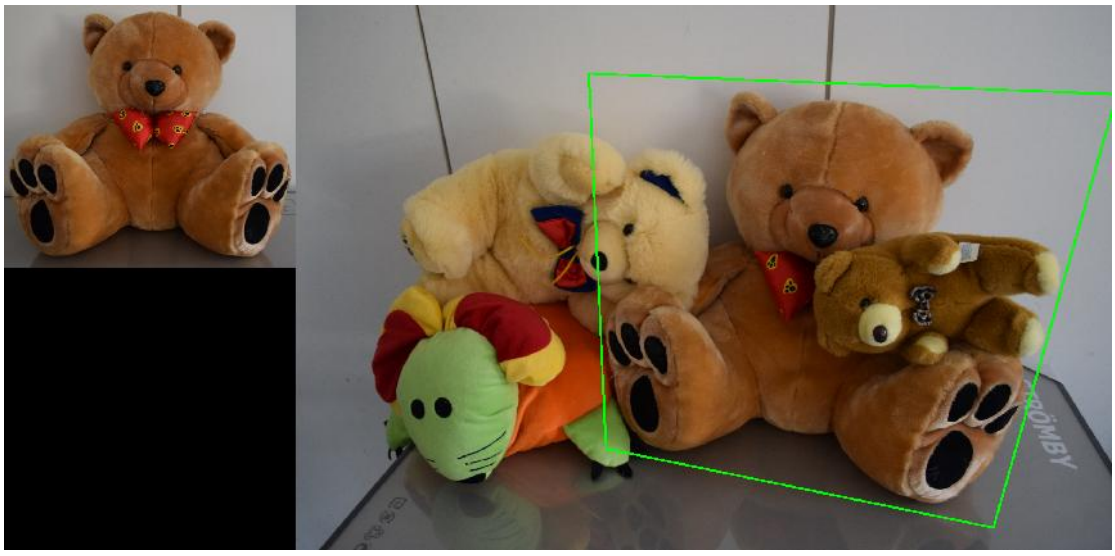

Figure 18: *Dataset 4*, the big bear with a red papillon keeping only the 10% of the best matches