

# Computer Vision and Pattern Recognition

Giacomo Boracchi, USI

due date: May 26, 2020

## General Rules

- This assignment is **due by May 26, 23.59 CET** to be uploaded on the icorsi platform.
- You are allowed to prepare your homework in **teams of two persons**, where both members are required to equally contribute to the solution of the whole assignment.
- The name of both members must be clearly written in your submission. Nevertheless, each student should submit their copy by their own (i.e., **double submission** mode: each student should upload their assignment) so that it will be easier for us to grade them.
- Upload a well commented Python script or (better) a **Jupyter Notebook**. You can also use Matlab if you prefer.
- Try to comment your design choice.

# Template detection: (10 pts + 5 bonus)

You are given the following target image (target1.png) and the following template (template1.png), and you are asked to implement an algorithm that identifies each instance of the template inside the target image. Once you have implemented and tuned your algorithm on these target-template pair, you have to repeat the test for the second setup which is characterized by more instances of the template in the image (target2.png and template2.png)

By solving this problem you have necessarily to:

- Implement a multi-model fitting strategy **via sequential model fitting**. The most straightforward option is to adopt Sequential Ransac, but you can in principle use any sequential extension of the algorithms shown during lectures.
- You are allowed to use findHomography function implemented in openCV -- **as long as you have full control over that** -- and you can perform the basic robust model fitting procedure described during lectures. You are also welcome to develop your own implementation of Sequential RanSaC (which might come handy for the project).
- After each detection, **estimate a pixel-wise mapping** between the template and each instance in the target image. Draw bounding boxes of the mapped template over the image.
- **Define a suitable stopping criteria to the sequential model fitting iterations**, namely when to stop searching for other instances of the template in the image.
- **Implement a criteria to discard bad-homographies** that might pop-up during the search. These are more likely to occur in the second template matching setup. Depending on how brilliant your solution is, you can get some bonus points.

Target1



Tmplae1



The expected final result for this kind of image is:



Target2



Template 2



Expected outcome

All objects identified

