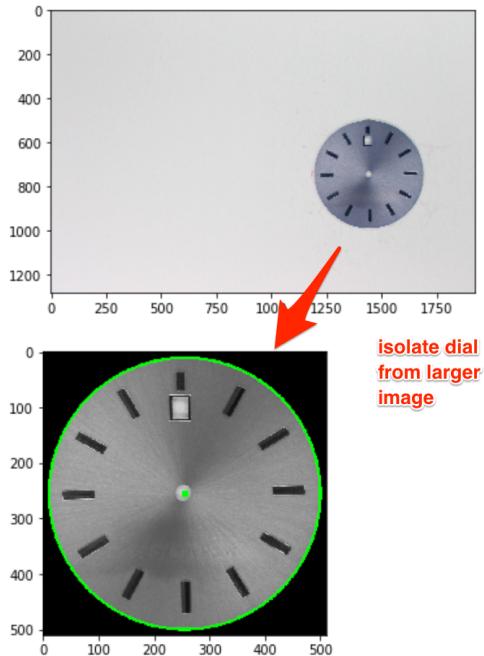


Work in progress overview report for feedback.  
(Packaging of script not completed yet)

**Program steps:**

- 1. Import image of standard, „good“ dial as a template for comparison.**  
Script crops out circular region.

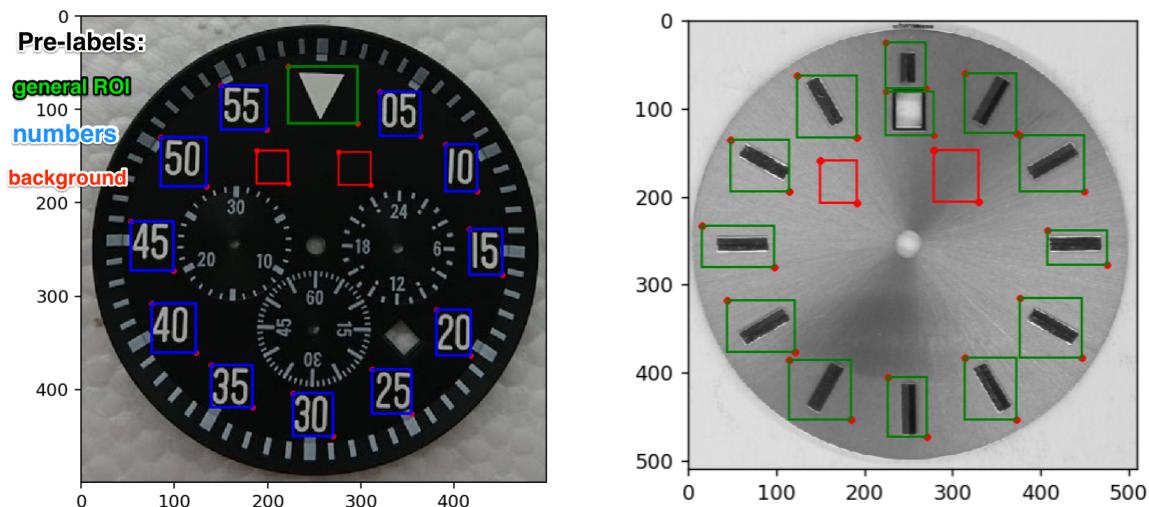


- 2. User labels regions of interest (ROI boxes) on template**

For each new type of watch dial, the user labels the following regions.

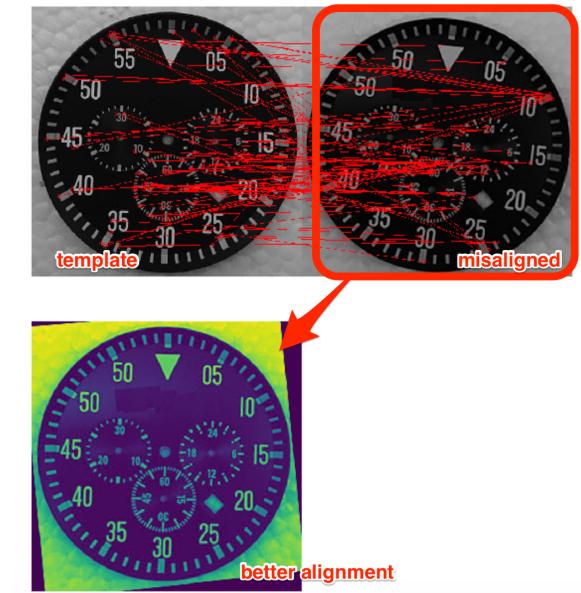
- General ROI (*green*; e.g. marks around the edge which may be missing in a faulty dial)
- Numbers (*blue*; in case wrong number gets stamped on dial)
- Background (*red*; this gives a reference color for background subtraction).

These coordinates are compared between the template and the factory photo of the dial.



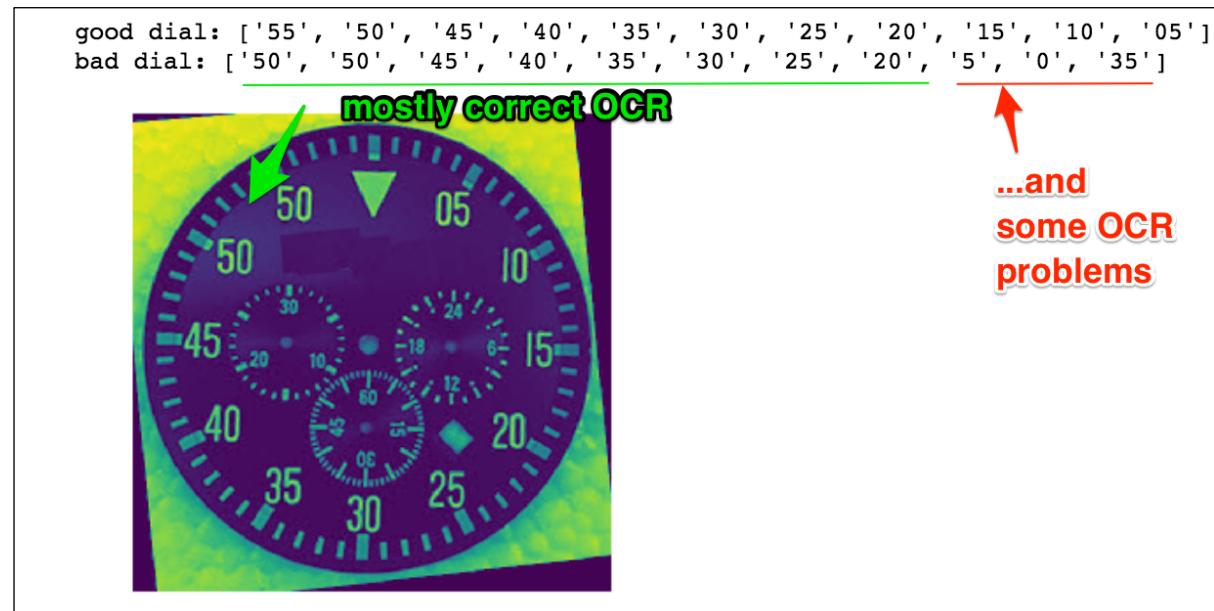
### 3. Factory images for comparison are imported and aligned with the template

Currently, this is done within the jupyter/python notebook using SIFT feature matching. For a final version, this would presumably be done directly by a script on the factory computer connected to the dial camera.



### 4. Numbers inside the ROI boxes from step 2 are classified and errors are flagged

This is done using a pre-trained neural network (relatively fast & accurate); the alternative is pytesseract (commercial OCR software, unfortunately quite slow and inaccurate). Accuracy seems to be around 90%.



**5. Non-numeric marks inside the “general” ROI boxes from step 2 are compared and major deviations from the template are marked.**

This is done using thresholding based on background ROI color; checking for differences in mean intensity; and then marking the image at the location of the deviation.

