**Part 4-1. Patch-wise keypoint detection (6%)**

Q1: Instead of detecting keypoints in a patch-wise manner, apply cv2.goodFeaturesToTrack to the entire image fence.jpg directly. Please append the result to this cell. (**Note that you need to use relative path and save your images in the submitted zip file, otherwise we cannot see your images.**)

Answer: <img src='results/fence-4-1-keypoints.png'/>

A close-up of a person's skin

Description automatically generated with low confidence

Q2: Compare the difference when you take N=(P⋅Q)⋅Np points for the entire image vs. considering Np points for each (P⋅Q) patch. Briefly describe the difference in outcome and explain why differences arise.

Answer: Computing keypoints over the entire image produces fewer and less dense keypoints throughout the image than patch-wise detection, with larger concentrations in some areas. Keypoint detection over the entire image is more affected by global elements in the image such as background (behind fence) and intensity.

Q3: What are the benefits of patch-wise keypoint detection for translation symmetry detection?

Answer: Allows isolation of patches that we want to detect translation symmetry for, enabling more accurate computation of local translations. The detection is not as susceptible to changes in background and global image intensity.

**Part 4-2. Clustering (6%)**

Part 1 groups the points first with mean-shift clustering and then refines with a K-means clustering.

Q1: For the first stage, explain the benefits of applying mean-shift instead of K-means.

Answer: <your answer here>

Q2: For the refinement stage, why do we discard the small clusters?

Answer: <your answer here>

Q3: Please briefly describe the relationship between the radius c and running time in the efficient implementation of mean-shift clustering.

Answer: <your answer here>

**Part 4-3. Affine transformation (3%)**

In Part 2, affine transformation and RANSAC are used to find proposals. In the function of get\_proposal(), only the points closest to a are transformed.

Q1: Why don’t we transform all points in the cluster to count the number of inliers?

Answer: <your answer here>