Homework 3 and 4

HW3: implement a *corner matching* algorithm based on template matching:

1. The input is two images I_1 and I_2, and the corner sets C_1 and C_2 detected in I_1 and I_2 using Harris Corner Detector (HW2).
2. Denote the $k \times k$ window patch centered at a corner f as $P(f)$.
3. Using zero-mean normalized cross correlation (ZNCC), you can measure the similarity score $S(P, P')$ between two patches P and P'.
4. For each $k \times k$ patch surrounding corners $f_i^1 \in C_1$, find its **most similar** patch from the patches surrounding corners $f_j^2 \in C_2$ using ZNCC, namely,
$$f_j^2 = argmin_{f_x^2 \in C_2} S\left(P(f_i^1), P(f_x^2)\right).$$
5. Make $F_i = (f_i^1, f_j^2)$ as a corresponding point pair if $S\left(P(f_i^1), P\left(f_j^2\right)\right) > \theta$ where $\theta = 0.8$ is a similarity threshold.
6. The output of your complete algorithm is a set of corresponding pairs $\left\{F_i = \left(f_{i_1}^1, f_{i_2}^2\right)\right\}$, where $f_{i_1}^1 \in I_1, f_{i_2}^2 \in I_2$.

HW4: implement an image panorama based on your HW3 result.

1. Following the 8-point algorithm, **compute a homography** H between I_1 and I_2 using the corresponding pairs$\left\{F_i = \left(f_{i_1}^1, f_{i_2}^2\right)\right\}$ you computed above, more specifically, making $f_{i_1}^1 = H f_{i_2}^2$.
2. **Compute Dimension of the Stitched Image.** (See the figure and explanation in the next page for implementation detail.)
3. **Stitch $\boldsymbol{I_2}$ to $\boldsymbol{I_1}$**, using the simple average blending. (See the explanation in the next page for implementation detail).
4. Your final output is the stitched image I_S.

HW3 and HW4 are due: 11:59pm, Apr. 17th and Apr. 23th, Respectively.

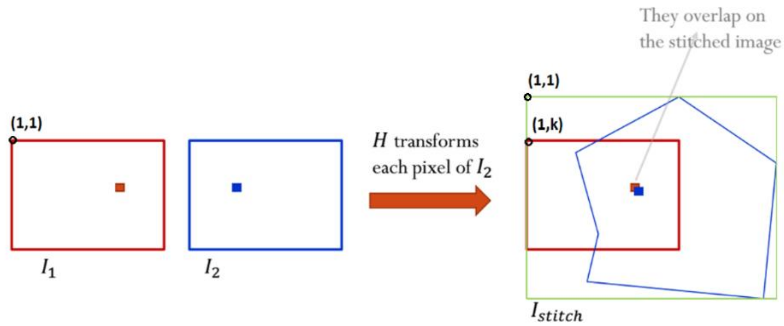
Submit your codes and results on Moodle.

If you are late for k ($k < 6$) days, your score will be calculated as:

Final homework score = (The score based on your codes) $\times$ 0.95^k

If you are late for more than 6 days, you get no score.

How to Find the Dimension of the Transformed Image $H(I_2)$ and the Stitched Image?



1. You need to first find the range of the transformed image I_2 under the transformation H. For each boundary pixel point $(u_i', v_i')^T \in B(I_2)$, compute its corresponding position (u_i, v_i) in the coordinate system of image I_1: $(\hat{u}_i, \hat{v}_i, \widehat{w}_i)^T = H(u', v'_i, 1)$, and then $(u_i, v_i) = (\frac{\hat{u}_i}{\widehat{w}_i}, \frac{\hat{v}_i}{\widehat{w}_i})$.
2. Compute the min and max x and y coordinates of transformed I_2 , and compare with the bounding box of I_1, you will get the dimension (green bounding box) of I_S.
3. Find a translation of the coordinate system of I_s , so that its new range $R(I_s)$ has the smallest u and v being 1. Denote this translation as T_s. Then $I_s = T_s(I_1 \cup H(I_2))$.

How to Stitch and Blend the Two Images?

Initialize I_s as a black image with the right dimension. For each pixel $(x_i, y_i) \in I_s$, compute its corresponding position in I_1, $p(u_i, v_i) = T_s^{-1}(x_i, y_i)$ and corresponding position in I_2, $p_i'(u_i', v_i') = H^{-1}\left(T_s^{-1}(x_i, y_i)\right)$.
Then:

(1) if both p_i and p'_i are outside their corresponding iamge range, leave it as black.
(2) if only p_i is within the parameter range of I_1, fill it using $I_1(u_i, v_i)$;
(3) if only p'_i is within the parameter range of I_2, fill it using $I_2(u'_i, v'_i)$;
(4) If both of these two pixels are within the range of their corresponding image, blend the grayscale value by choosing an average of grayscale value, $I_s(x_i, y_i) = (I_1(u_i, v_i) + I_2(u_i', v_i'))/2$.

Note 1: we use $I_1(u, v)$ to denote the grayscale value in image I_1 on pixel location $(u, v)^T$, same for I_2 and I_s.
Note 2: if (u'_i, v'_i) does not have integer value, you should do a bilinear interpolation to get $I_2(u'_i, v'_i)$ from its neighboring integer pixel's grayscale values.