Class Notes on Project 2

Use MATLAB corner detector ⟶ Harris

- Get corner responses with cornermetrix
- Find a threshold to reduce points
- Use imregionalmax() to get local maxima

Adaptive Non Maximal Surpression

- Reduce number of points
- Get even distribution across Image

Create Feature Descriptors

- Take local region around point
- Blur & Downsample *(Does MATLAB have a downsample function? i.e. 40x40 - 8x8)*
- Flatten to 64x1
- Set mean of pixel values to be 0 and variance to be 1
- These steps make it more robust to slight variation in appearance (illumination,etc.)

Match Features

- K-Nearest Neighbor search between feature descriptors (Use MATLAB function - *knnsearch()*)
- Bad Matches ⟶ Outlier Rejection

Outlier Rejection

- Ratio of match qualities
- Take top 2 matches and threshold the ratio between the two (SSD)
- Use RANSAC to further reduce outliers
    1. Select 4 feature pairs
    2. Computer Homography (exact)
    3. Computer inliers where SSD(pi, Hpi) < thresh
    4. Keep largest set of inliers
    5. Re-compute least squares H estimate on all inliers
- Reduces to few to no outliers

Blending

- Find overlap
- Take average of pixel values
- Better way
    1. Take distance of each pixel from seam
    2. Weight blending based on distance from seam

Warping Panorama

- *What if the images aren't in order?*
- Match all images to eachother and pick two with the best matches
    1. Select one image as the center
    2. Warp other images towards it, one-by-one
        - *Why warp images towards the center?* ⟶ Less error accumulation
    3. Blend images together
- Hints on warping
    1. imwarp() will keep first image's size ⟶ image will be cut off
    2. Solution: Find out dimensions of ifnal panorama and min/max x/y of each image in panorama
    3. Helpful Matlab Functions/Classes
        - projective2.outputLimits
        - Imref2d ⟶ Can be passed to imwarp() to describe coordinate transform