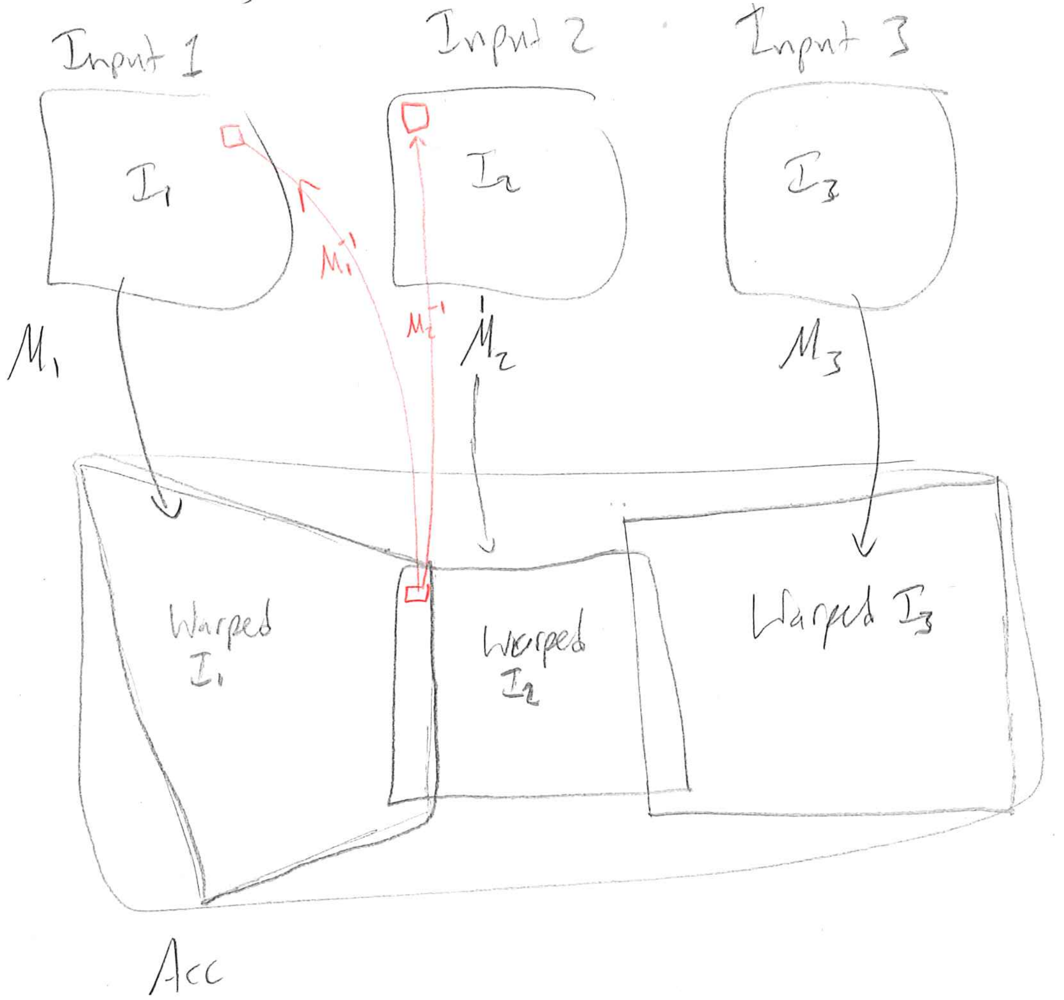
Panorama Stitching



Todo 8 and 9 find the necessary size of Acc so it fits all the images, and adjusts the Transforms (M) to map all images into the common Acc coordinate system.

Todo 10 warps a given image and adds it to the accumulator, using inverse warping: ← see also L09 slides 34-39

for each $\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$ in the relevant part of acc, compute input image coordinates:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = M^{-1} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$

need to divide by w to make third coordinate 1 (normalize)

. Since $\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$ is not an integer pixel location, Use bilinear interpolation to determine Source pixel value, then fill into Acc[y, x].

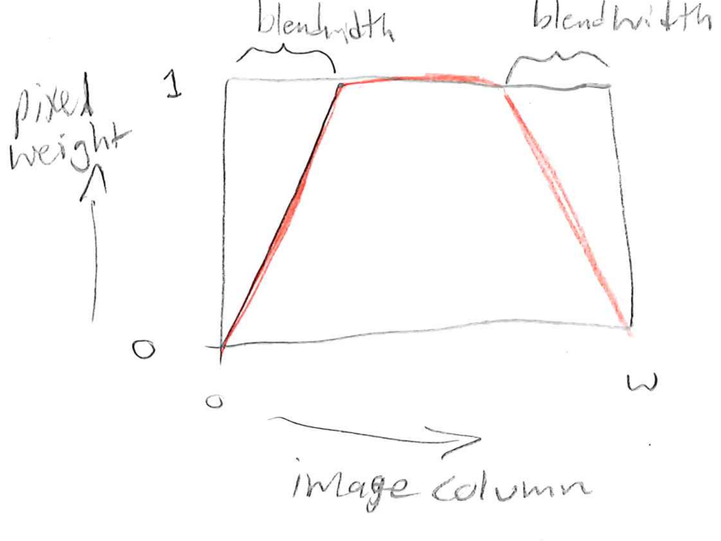
Start with: just round $\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ to integers to do nearest neighbor interpolation and ignore blend width.

For proper blending, pixels need to get added to Acc with associated weights. If a pixel in Acc has multiple images overlapping, the final pano value is

$$\frac{\alpha_1 p_1 + \alpha_2 p_2 + \cdots + \alpha_K p_K}{\sum_i \alpha_i}$$

i.e. the weighted average of pixel values, normalized so total weight = 1.

The weights are determined by blendwidth:



The accumulator collects total weight in its 4th (alpla) channel.

Todo 11 simply divides acc by the total weight in the alpha channel.

Tips:
- Lecture 9 has sample code for inverse warping.
- Make each input image 4 channels with 1 in 4th channel, then scale pixels according to the blending function before interpolation. Then, acc gets $+\alpha_i p_i$ in the first 3 channels, and $+\alpha_i$ in the 4th.
- Start by just overwriting values in Acc, then worry about blending and feathering after the inverse warping works.