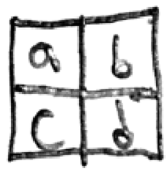
HW1 Answer Key
CMSC426, Spring 2018
Professor Yiannis Aloimonos,
TAs Jack Rasiel and Kaan Elgin

Q1:

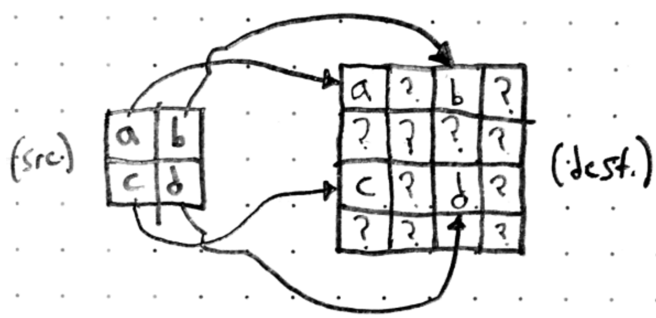
One problem with forward warping occurs when the destination (warped) image is larger than the source image. Consider a toy example, a 2x2 source image:



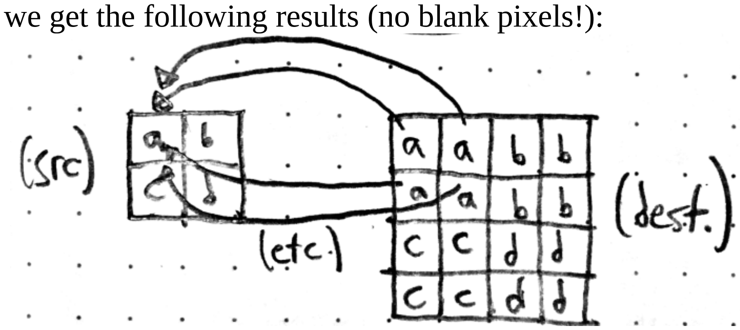
And a homography H which is a simple 2x scaling:

$$H = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Using forward warping, we get the following results (note the unassigned pixels!):



With inverse warping, we get the following results (no blank pixels!):



Q2, a:

We'll use the following formula:

e = probability that a point is an outlier
s = number of points in a sample
N = number of samples (we want to compute this)
p = desired probability that we get a good sample

Solve the following for N:

$$1 - (1 - (1 - e)^s)^N = p$$

Let's take a moment to derive it:

- *e* is p(outlier), so *1-e* is p(inlier)
- thus *(1-e)^s* is the probability of choosing four points that are all inliers.
- 1 - *(1-e)^s* is the probability of choosing four points that are NOT all inliers (i.e. one or more outliers).
- (1 – (1-e)^s) ^N is the probability of choosing four points N times, and having every set of points include one or more outliers.
    - FINALLY: **1 - (1 – (1-e)^s) ^N** is the probability of choosing four points N times, and having one or more sets of all inliers.

For this question, (1-e)^s = $\mathbf{p_g}$.
So, $\mathbf{p = 1 - (1 - p_g)^N}$

Q2, b:

We'll use the formula from 2A again here.
In this problem, e = .5, *s* = 4, and *p* = *.95*.

- Plugging into the formula, we get: $1-(1- (1- .5)^4)^N = .95$
- Simplify and solve for N:
    - $1 - (1 - (.5^4))^N = .95$
    - $1 - (1 - .0625)^N = .95$
    - $(0.9375)^N = 1 - .95 = .05$
- **N = log(.05) / log(0.9375) = 46.417739680899224 ~= 47**
    - so, **47 iterations**

Q2, c:

In this problem, we have two images, each of which contains 100 feature points*.

To brute-force the point matches, we need to pair every possible set of 4-points in each image with each other. This is a combinatorics problem:

- For each image, there are 100c4 sets of points.
- Thus there are (100c4)^2 pairings of the sets from both images.
- For each pair of 4-point sets, there are 4! ways to match up their points.

    - Thus the **total possible pairings is (4!) * (100c4)^2 = 24 * (3,921,225)^2**
- More important than the exact number of pairings is the overall complexity of brute-forcing the problem.
    - 100c4 is $\frac{100!}{4! \times 96!}$, which simplifies to (100 * 99 * 98 * 97) / 4!   Based on the numerator, nCk is O(n^4).   Since we square 100c4 in our problem, **brute forcing has a complexity of O(n^8)**. Yeesh!

Let's compare that to RANSAC. Let's assume that we want p = .9999 of getting an outlier-free sample:

- We can use the following formula, derived from the formula given in Q2A:

$$N = \log(1-p) / \log(1- (1-e)^4)$$

- With p = .9999,   N ~= -9.21034 / log(1- $(1\text{-}e)^4$)
- The number if iterations is now a function of *e*, the probability of a point being an outlier.
    - If *e* is really bad, say .9 , then N ~= 90000. If e is really good, say .2, then N ~= 17.
- In practice *e* is rarely as bad as .9. But taking it as a worst-case scenario: **RANSAC gives at least a five order-of-magnitude improvement!**

***Note: in this question, the feature points have not been matched between images.** Many students assumed that the points were already matched. This mistake was widespread enough that we are not deducting points for answers made based on it (so long as there weren't other errors in your answer).

Q3

Four point correspondences are required to establish a homography. This is because the linear system has eight degrees of freedom. For full credit, we expected you to explain why this is. One good explanation is in terms of the constraints of the linear system:

A homography is a 3x3 matrix, so we are solving for its nine elements. However the system has eight degrees of freedom, not nine. This is because a homography is only defined up to some scale factor. This reduces the degrees of freedom to eight. Since there are eight degrees of freedom, we need at least eight equations. There are two equations for each point correspondence, so we need at least four points.