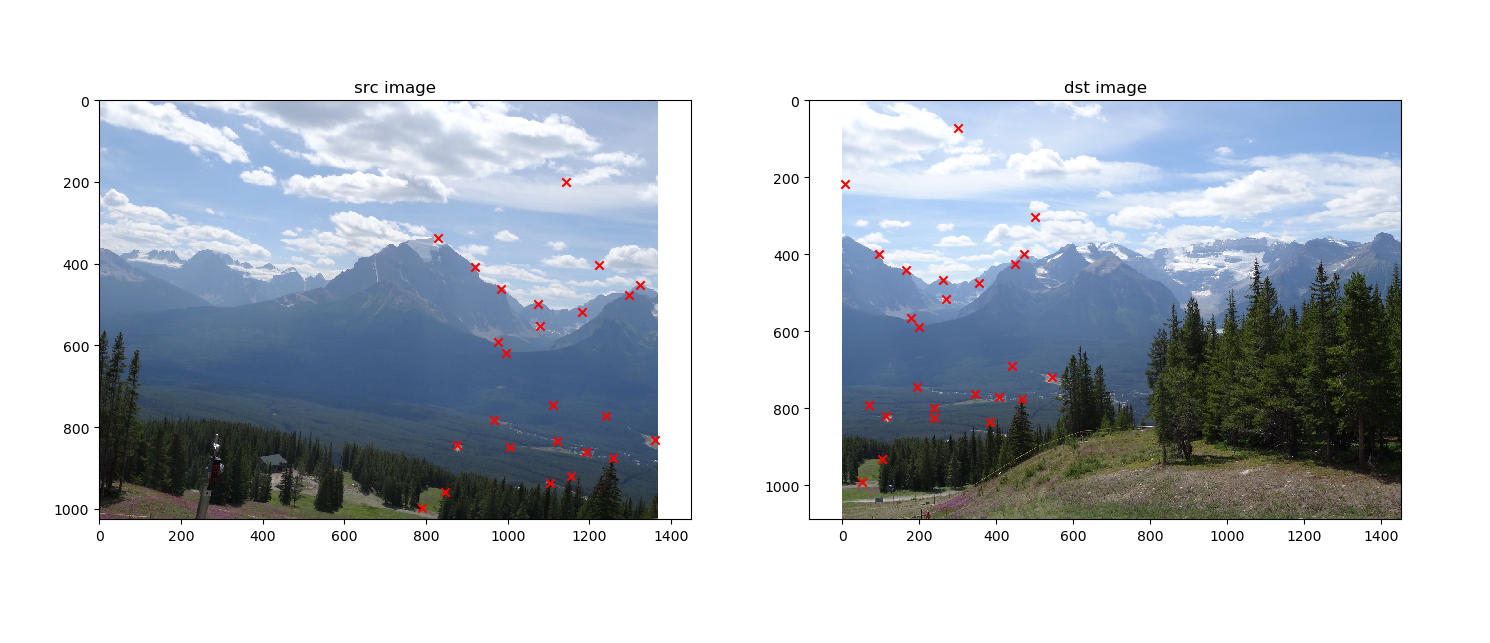
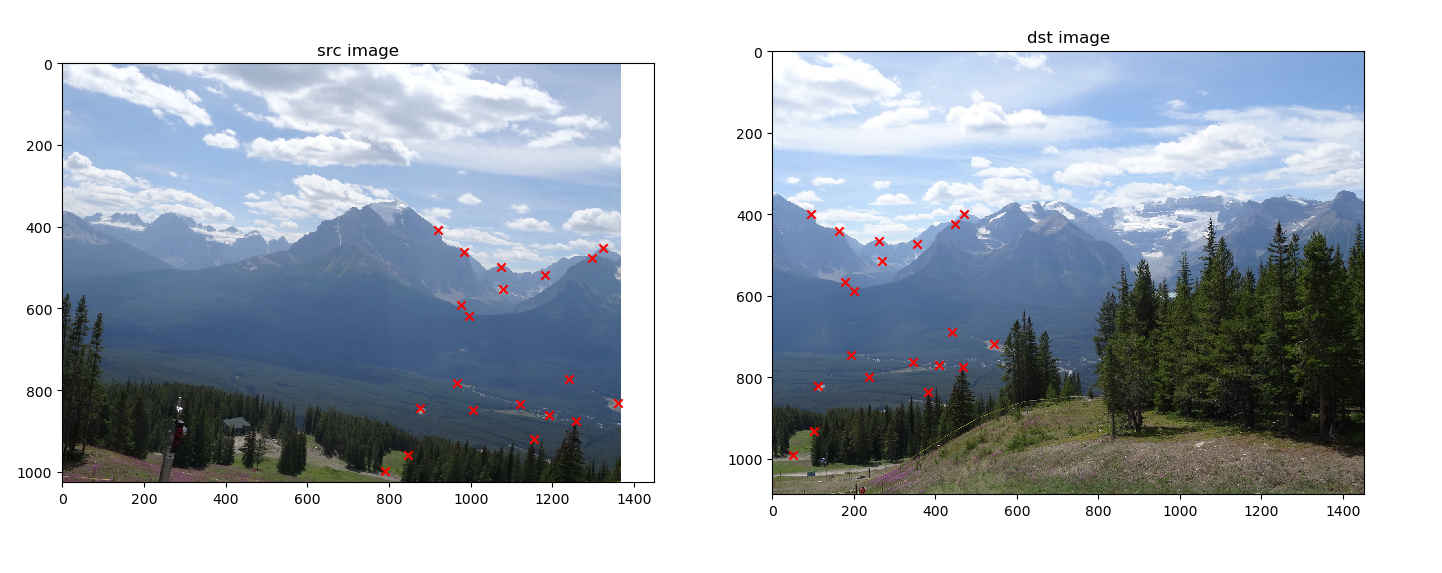
**Computer Vision - Homework 2 solution**

Here are the images with all the matching not perfect points:

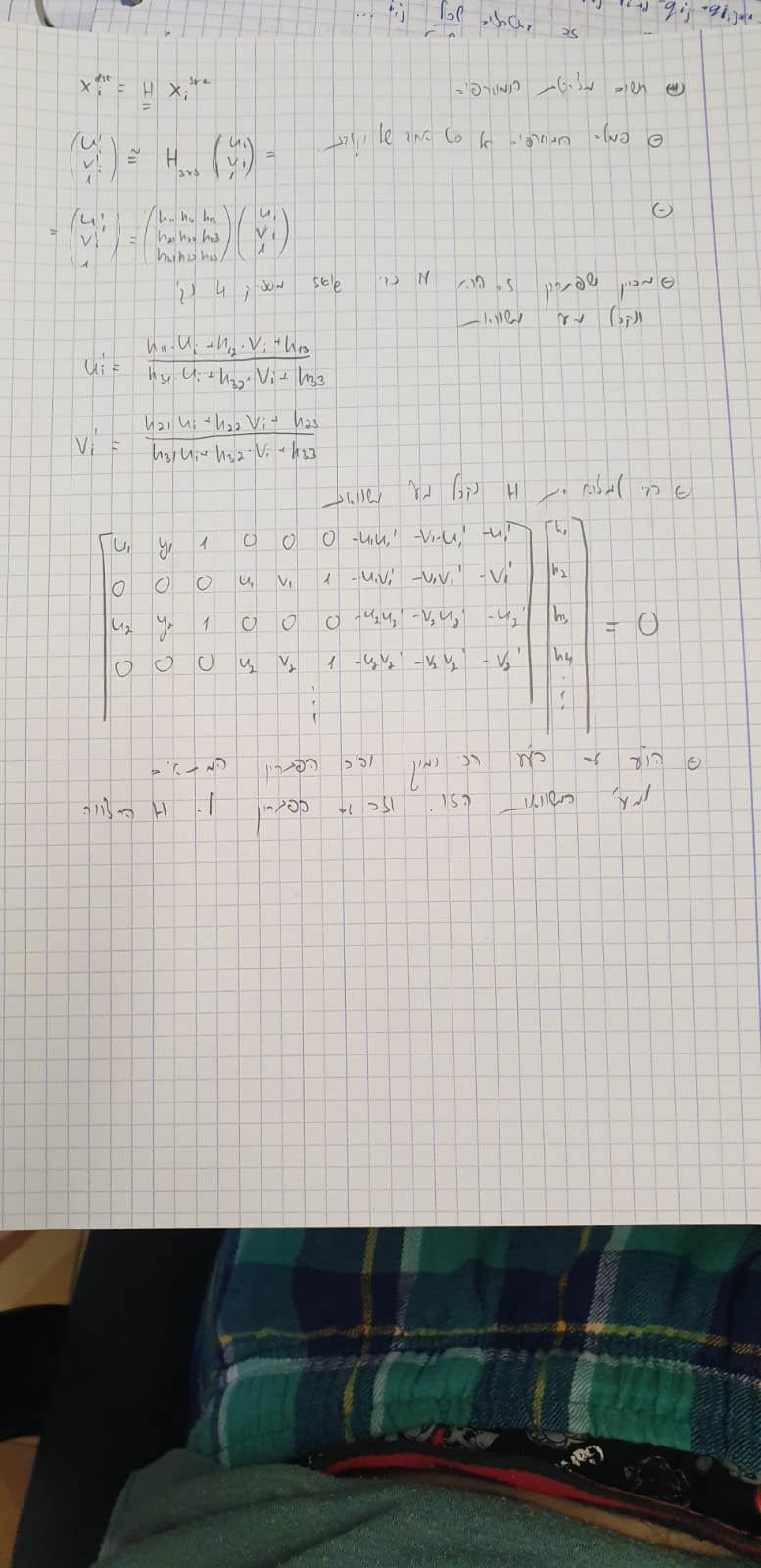


And the matching perfect points:

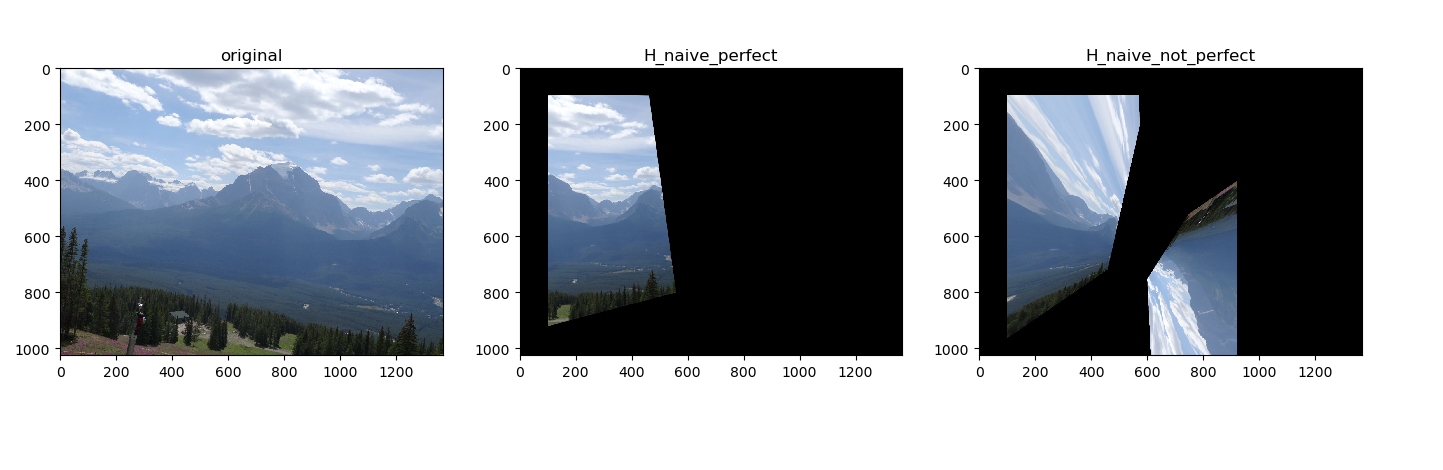


**Part A**

1. Here is the development:



4. Here are the results for the perfect and not perfect matching set of points:



5. The problems with forward mapping are:

* There is no information of the picture beyond its boundaries. When we are doing forward mapping the result image won’t be a perfect rectangle because the transformation don’t know how to complete it.
* There is a need to perform an interpolation in a discrete image. In this case the image will be slightly different from the original one.
* The result image of the naive transformation with the not perfect set of matching points give bad results in comparison to the one with the perfect set of points. This is due to no robustness to noise and mismatched points.

6. Our answer to this question is in the previous answers.

**Part B**

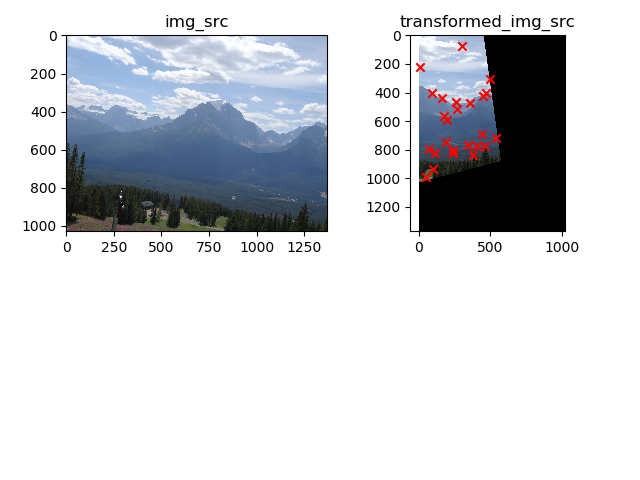
9.

* It is given that for using homography we need 4 points to calculate a model, therefore n=4.
* It is also given that inlier percent is 80% - meaning w = 0.8.
* In class we showed that
* For desired success rate of p = 0.9 we will get:
* For desired success rate of p = 0.99 we will get:
* In order to cover **all** the options, we will need almost an infinite amount of random points – as the points are selected randomly from the dataset (That is, if we used replacement. If we do not use replacement, so every point drawn can’t be drawn again, our number of iterations is dependant of the dataset size (and not just the percentage of inliers).

10. When running the original src & dst images with matches.mat, we get the following results:

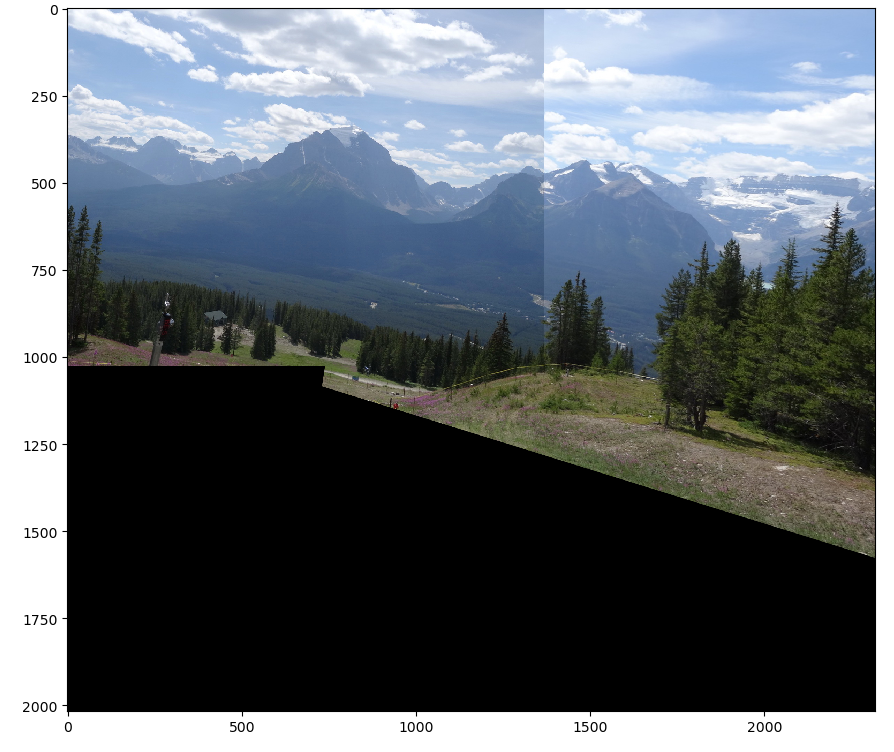
* W =

|  |  |  |
| --- | --- | --- |
| -0.001123 | -0.000164 | 0.999919 |
| -1.05e-05 | -0.001054 | 0.012562 |
| -2.97e-07 | -4.35e-08 | -0.000783 |

* Source image after projective transform (forward mapping):  
  
* Comparing the results to sections 4 & 6 we see:
  + Using RANSAC on the matches dataset gives very similar results to the transformation we get from the perfect matching points set.
  + Using RANSAC does not result in image artifacts, because the way it filters out the outliers match points.

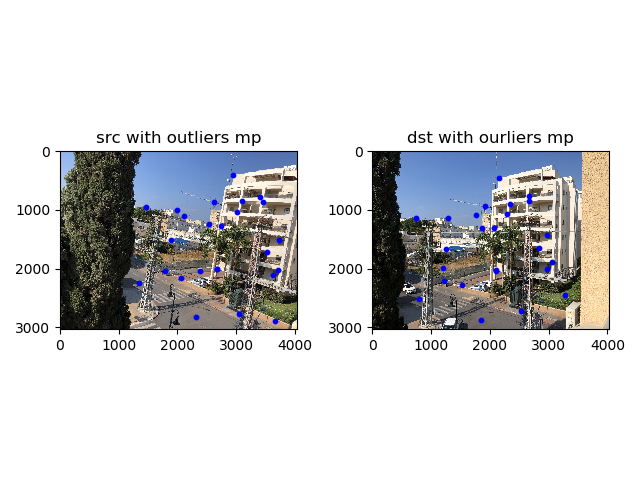
**Part C**

13. This is the panorama as asked with the matched points and ransac homography:

****

14. Using a pair of our own images:

* Images with the matching points (25 total, 4 “wrong”):



* Output panorama image:  
  