Exercise in Feature Matching

Henrik Aanæs, `haa@imm.dtu.dk`

February 4, 2013

This is the exercise in feature matching for course 02501. This exercise covers/illustrates the material of Chapter five of my lecture notes, on builds on the previous exercise on feature *extraction* and on the Camera Model. In this exercise you are expected to write the code for matching Harris corner's via correlation, experiment with it under different image conditions. Following this you should download code for using the SIFT detector, and compare it to matching Harris corners via correlation. Lastly, you should integrate the use of epipolar geometry in the feature matching.

The images you are expected to work with are:
`House1.bmp`
`House2.bmp`

And later the images in `TwoImageData.mat`, which should be located together with this text.

1 Matching via Correlation

Here you are to match the corners between two images via correlation. To do this it will be an advantage to write your own functions in MatLab, as opposed to just using a script. If you are not familiar with doing this try typing `helpwin function` in MatLab. This problem can be split up into the following tasks:

1. **Task:** Write a function which extracts a $(2n + 1) \times (2n + 1)$ pixel patch from an image. This patch shall be centered at a given point x, y.
2. **Task:** Enhance the above function such that it returns a $(2n + 1) \times (2n + 1)$ array of zeros if x, y is not at least n pixels from the boarder of the image.
3. **Task:** Write a function that calculates the cross-correlation between two given image patches.
4. **Task:** Use the Harris corners extracted from the two house images above. Make a function, that for a given corner in `House1.bmp`, finds the corner in `House2.bmp` with the highest correlation.
5. **Task:** For each feature in `House1.bmp` find the best match in `House2.bmp`.
6. **Task:** Do the reverse, i.e. for each feature in `House2.bmp` find the best match in `House1.bmp`.
7. **Task:** Extract the feature pairs which are each others best match.
8. **Task:** Illustrate the result.

2 Modifying the Problem

To investigate how correlation based feature matching degrades with a harder problem, you should here gradually modify `House2.bmp` and rerun your matching algorithm from above. You should modify the image via a homography, which you can do via the following code:

```
a=20*pi/180;
s=sin(a);
c=cos(a);
H=[c s 0;-s c 0;0 0 1];
%Note That H has to be transposed here if column vectors are used for
points
Tr=maketform('projective',H');
WarpIm=imtransform(im2,Tr);
```

Where the image is rotated 20°. Try for increasing angels of rotation, i.e. for `a` increasing from zero with appropriate steps.

- How does the performance of your feature matching depend on the rotation?

Also try the homography computed by

```
s=0.1;
H=[1 0 0; s 1 0; 0 0 1];
```

for different values of `s`.

-What does this homography do?

- How does the performance of your feature matching depend on s?

3 Matching with SIFT

Download code for SIFT matching from `http://www.vlfeat.org/index.html`. Get this code up and running and repeat the experiments presented above with SIFT features and descriptors.

- How does the results compare?

- Specifically, how do the results compare when increasing the deformation of the second house image?

4 Using Epipolar Geometry

In the exercise previous exercise "Exercise in the Camera Model", you computed the fundamental matrix between the two of the images from `TwoImageData.mat`. Repeat the experiments from above - omit the image warping. No enhance your feature matcher by only allowing matches which are consistent with the epipolar geometry.

- What is the effect of using epipolar?

Hint: Use the sampsons distance equation (2.40) in the lecture notes to compute if two features are consistent with the epipolar geometry. Set a reasonable noise level yourself.