Advanced Vision Practical 2

Bob Fisher
School of Informatics

February 2013

Abstract

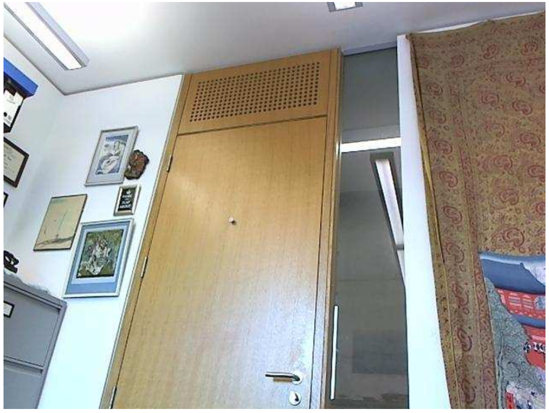
This describes the second assignment for assessment on Advanced Vision. The main goal is to reliably detect and track 3 coloured balls being juggled in a set of video data. The assignment is due: **4pm Thursday 28 February**. You must do this practical in teams of 2, and submit 1 PDF report only.

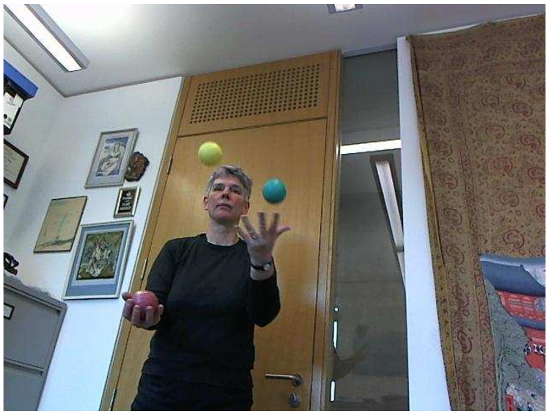
Task Background

At the URL:

`http://homepages.inf.ed.ac.uk/rbf/AV213`

you will find a background image `background.jpg` and a tar file `juggle1.tar` of 99 consecutive images. The background image and frame 2 are:





This assignment is about detecting and tracking the three coloured balls. (Note: if you are colour-blind, please find a partner who is not.)

There is also a file called `gt1.m` that has the estimated centres of each of the three balls in the format:

```
gt1 = [ ...
     [1 rr rc gr gc yr yc]', ...
     [2 rr rc gr gc yr yc]', ...
     so on
     [99 rr rc gr gc yr yc]' ];
```

rr/rc are the row and column coordinates of the red ball centre. gr/gc and yr/yc are the same for the green and yellow balls. These positions were measured by hand and are the 'ground truth' for this dataset.

Finally, there is a file `show_gt.m` that shows the 99 images one at a time with the ground truth overlaid.

Your task for this assignment is to write a set of programs that can:

1. detect the moving balls against the complex background,
2. compute a trajectory for each ball through all frames and
3. evaluate the correctness of the tracking against the ground truth dataset.

Each of these is described in more detail below.

Ball Detector

The image data is a set of RGB colour images and the camera is fixed. There is a background image and the lighting is mostly constant. You should be able to easily detect the balls using background subtraction. Threshold the detected regions.

Unfortunately, there are several problems that mean that the balls are not immediately obvious:

- Juggler's clothing: exploit the fact that this is very dark to remove these pixels
- Juggler's skin: much face detection is based on the fact that skin colour is quite distinctive, especially in normalised RGB. Look for papers on skin detection for ideas on how to use the unique colour to remove these pixels.
- The shadows and highlights on the door, etc: investigate how to do background subtraction using the normalised RGB images, where these effects are largely removed.
- The average lighting in the image is different between the live images and the background image. Maybe using the normalised RGB images would overcome this problem?
- Use the colours of the balls (from a sample training image?) to define a classifier for red, green and yellow pixels. Maybe normalised RGB is better for this?

Use operations like open/close or dilate/erode to clean up the image. Use the largest connected components. (See the IVR/AV notes for examples of these.)

Label the balls by their colours.

Compute the centre of mass of each detected ball.

Tracking

Determine the correspondence between the red ball in each image. The balls will not move much between consecutive frames so this should be easy and the Condensation tracker discussed in lecture is probably unnecessary. Maybe looking for the closest object of the same colour and about the same size would work? Repeat for the green and yelow balls.

Create 3 images showing the trajectories of each ball's centre of mass against the background image.

Evaluation

The ground truth file shows the ball centre as found by hand, for each ball colour. Compare your estimated ball centre (use the centre of mass of the detected pixels) with the ground truth centre for each ball and each frame (297 comparisons in total).

Report the number of balls that were detected within 10 pixels of the ground truth centre.

Draw an image with the detected ball centre as a black "+" and the ground truth ball as a black circle. Pause between each frame.

Your Report

Write a report that describes:

- The algorithms that you used for each stage of the process.
- How well the algorithms performed on the supplied test data. Show the statistical results described above.
- Show example images of each processing stage, including a few examples of detected foreground, the clothing, the skin and the balls.
- Show the trajectories of the 3 balls.
- Show examples of successful and unsuccessful detections.
- Report the detection statistics.
- Discussion on performance: successes and failures, causes of failures and potential remedies.
- Your code. Do not include code that was downloaded from the AV or IVR web sites.

Other Comments

1. You can use the lecture example code from:
   `http://www.inf.ed.ac.uk/teaching/courses/av/MATLAB/`
2. Because there are a limited number of MATLAB Image Processing library licenses available, use alternative MATLAB functions from
   `http://www.inf.ed.ac.uk/teaching/courses/av/MATLAB/UTILITIES/`

Assignment Submission

Submit your report in PDF online by 4pm Thursday February 28. The online submission line is:

```
submit av 2 FILE
```

where `FILE` is the name of your file.

Live Demonstration

There will also be a demonstration session assigned on the morning of Friday March 1, where you will have to demonstrate your code on a new dataset (taken also at the same time as the other images, including the same background). We'll email you about the location and schedule.

You will need your matlab program to show:

1. The detected foreground balls for each frame overlaid over the original images using the cross and circle method described above. Use a 1 second pause between images.
2. The trajectories of each ball overlaid on the background image (use one image for each ball).

Note that there is no ground truth file for the live demonstration.

The assignment is estimated to take 10 hours coding/test and 5 hours report writing per person, resulting in a 5 page report plus the code appendix. You must do this assignment in teams of 2. You must find your partner and email Bob Fisher (rbf@inf.ed.ac.uk) the name of your partner. A single, joint, report is to be submitted. Split the work so that each partner does some independently (i.e. share the work rather than duplicate it).

The assignment will be marked as follows:

Issue	Percentage
1. Clear description of algorithms used	30%
2. Performance on supplied image data	20%
3. Clear Matlab code	20%
4. Discussion of result quality and causes of any failures	10%
5. Performance on supplied demonstration image data	20%

Plagiarism Avoidance Advice

You are expected to write the document in your own words. Short quotations (with proper, explicit attribution) are allowed, but the bulk of the submission should be your own work. Use proper citation style for all citations, whether traditional paper resources or web-based materials.

If you use small amounts of code from another student or the web, you must acknowledge the original source and make clear what portions of the code were yours and what were obtained elsewhere. You can ignore this condition for the AV lecture examples, which can be used freely.

The school has a robust policy on plagiarism that can be viewed here:

`http://www.inf.ed.ac.uk/teaching/plagiarism.html`

The school uses various techniques to detect plagiarism, included automated tools and comparison against on-line repositories. *Remember: a weak assignment is not a ruined career (and may not reduce your final average more than 1%), but getting caught at plagiarism could ruin it.*