Motion

Dr. François Pitié
Electronic and Electrical Engineering Dept.
pitief@tcd.ie www.sigmedia.tv

This lab accounts for 5% of the final 4C8 mark.

Report Instructions: You are required to compile a report that answers specific instructions included in the report. Include relevant code and output figures in your report.

Submission Requirements: Labs Reports should be typed up and submitted electronically using the **PDF file format** via the module's blackboard page by the specified deadline. Remember to put your name and student number on the top of your reports.

Plagiarism Policy: Any submitted code or answers that can be seen to be plagiarised from other sources will result in 0 marks being awarded for that section of the lab and may result in you being awarded 0 marks for the entire assignment. Anti-plagiarism software will be used on submitted materials.

Introduction

The aims of this laboratory are as follows

1. To use motion vectors in prediction and processing of images
2. To implement a temporal noise reduction algorithm
3. To observe how motion compensated processing differs from non-motion compensated processing
4. To evaluate the difference between IPPP and IBBP schemes for prediction

Test images can be found in `y:/image_stills` `y:/sequences`. Raw video for restoration can be found in `y:/sequences/dirty`. All data without `.bmp; .jpg; .mpg etc` extensions are stored in RAW BINARY FORMAT. Grey scale images are 8 bits per pixel (each pixel is stored as **Unsigned Chars**). See the read me files in each directory to work out how big the images are. Ask a demonstrator where the data is located in case you have a different set up.

1 Block Matching

The Matlab file `block_matching.m` contains the skeleton of a Block Matching algorithm with just the error calculation missing. It uses a full search strategy.

Q 1.1 *Edit the file to add the lines required to find the block with minimum MAE DFD and it to* `mc_block` *and assign the offset to* `motion_x` *and* `motion_y`. *Report relevant code into the report. You can test your code using the* `testBlockMatching` *unit test function.*

2 Motion Compensation Error

The data that you will use initially is in `qonly.360x288.y`. It is a sequence of frames from a Bond movie stored as raw data. Each pixel is 8bit, and each frame is stored one after the other sequentially in a binary file. Each frame is stored in the same way as a raw image would be.

It is typical to evaluate the performance of a motion estimator by plotting the average motion compensated error in some way versus the frame number. The motion compensated frame difference at each pixel site is already stored for you in `dfd`, noting that the edges are ignored. Each entry in that matrix therefore shows $e(h,k) = I_n(h,k) - I_{n-1}(h + v^x_{n,n-1}(h,k), k + v^y_{n,n-1}(h,k))$ at each site (h,k). Here the motion vector mapping position (h,k) in frame n into the previous frame $n-1$ is $(v^x_{n,n-1}(h,k), v^y_{n,n-1}(h,k))$, where the horizontal component of motion is $v^x_{n,n-1}(h,k)$ and $v^y_{n,n-1}(h,k)$ is the vertical component. The MAE over the whole frame between frames n and $n-1$ is then

$$E_{n,n-1} = \frac{1}{HK} \sum_{h=0}^{h=H-1} \sum_{k=0}^{k=K-1} |e(h,k)| \tag{1}$$

Q 2.1 *Plot the motion compensated MAE per frame for the first 30 frames into the report. On the same graph, report the non-motion compensated MAE per frame for the whole sequence.*

3 Analysis

Q 3.1 *By considering your results point out the regions in the images where Block Matching works well and where it does not. Explain your findings and point to the relevant original frames and DFD's.*

Q 3.2 *Try changing block sizes, motion threshold and search with w to improve the MAE plots for* `qonly.360x288.y`. *Graph a few combinations you try and rank them in terms of their MAE performance and computation times. How do they affect the resulting MAE and motion field?*