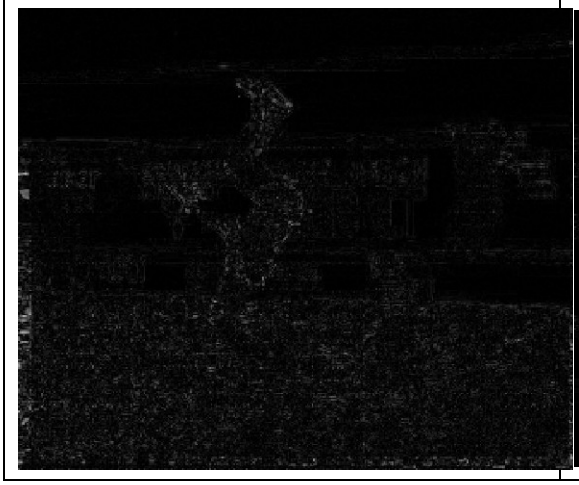
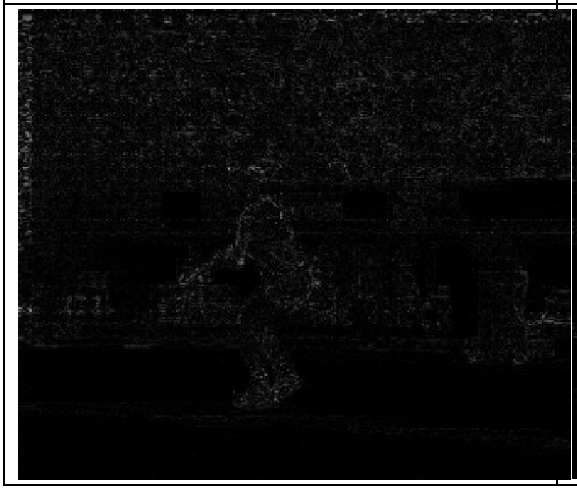
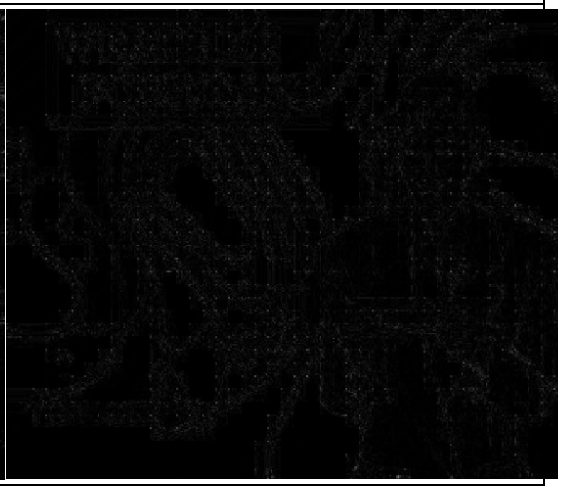
程式執行硬體環境：

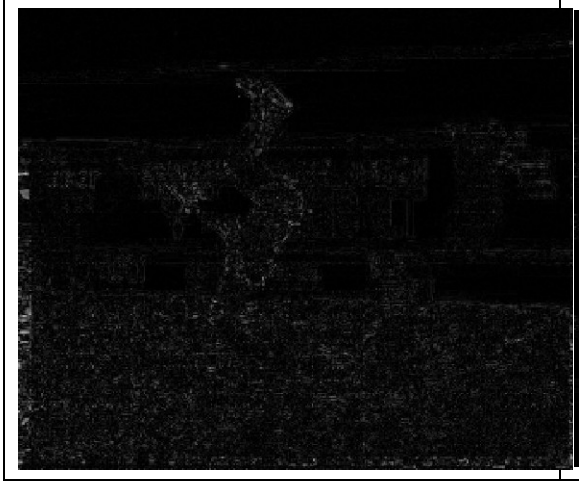
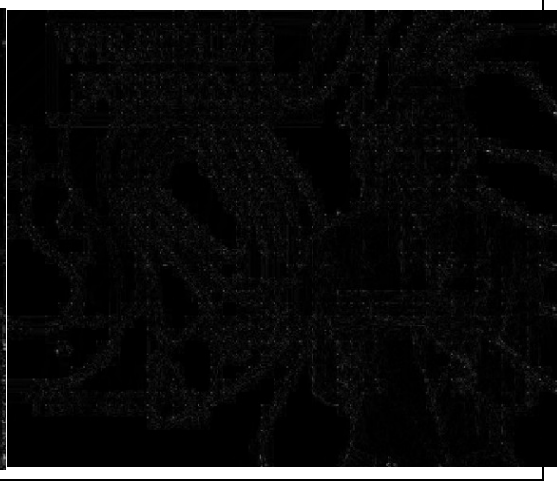
程式的部分承襲 HW3 的物件繼續實作，新增了 ME 計算 MV，MC 用 MV 預測重建影像，SubImage 採類似 OpenCV 的方式相減影像，ADDImage 採類似 OpenCV 的方式相加影像，GOP = 16。

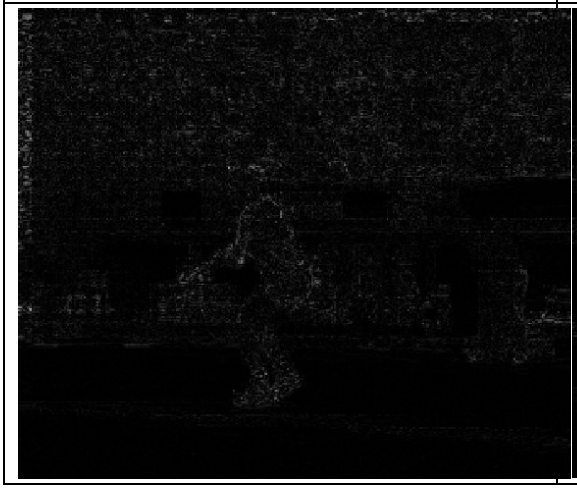
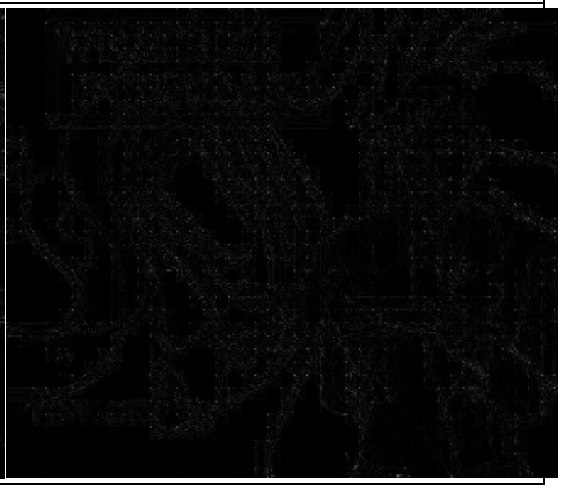
```
62    for(int i = 0;i < compression.TotalFrame();++i){
63       std::cout << "frame" << i << std::endl;
64       clock_t time = std::clock();
65       std::vector<std::vector<unsigned char> > origin_image;
66       std::vector<std::vector<double> > dct;
67       compression.NextFrame(video,origin_image,i);
68       if((i % GOP) == INTRA){
69          WriteFrame2File(diff_file,frame_buffer);
70          compression.OneD_Block_DCT(origin_image,dct);
71          compression.Quantization(dct,INTRA);
72          compression.IQuantization(dct,INTRA);
73          compression.OneD_Block_IDCT(dct,frame_buffer);
74          WriteFrame2File(res_file,frame_buffer);
75          double PSNR = compression.PSNRComputing(origin_image,frame_buffer);
76          std::cout << "PSNR : " << compression.PSNRComputing(origin_image,frame_buffer) << " dB" << std::endl;
77          psnr_file << PSNR << std::endl;
78       }
79       else{
80          compression.ME(origin_image,frame_buffer,d,mode);
81          compression.MC(frame_buffer,predict_frame);
82          WriteFrame2File(rec_file,predict_frame);
83          compression.SubImage(origin_image,predict_frame,diff_frame);
84          WriteFrame2File(diff_file,diff_frame);
85          compression.OneD_Block_DCT(diff_frame,dct);
86          compression.Quantization(dct,INTRA);
87          compression.IQuantization(dct,INTRA);
88          compression.OneD_Block_IDCT(dct,res_frame);
89          WriteFrame2File(res_file,res_frame);
90
91          compression.AddImage(res_frame,predict_frame,frame_buffer);
92          double PSNR = compression.PSNRComputing(origin_image,frame_buffer);
93          std::cout << "PSNR : " << PSNR << " dB" << std::endl;
94          psnr_file << PSNR << std::endl;
95       }
96       double diff_time = (double)(clock() - time) / CLOCKS_PER_SEC;
97       std::cout << "time : " << diff_time << "s" << std::endl;
98       time_file << diff_time << std::endl;
99    }
```

1. absolute difference sequences

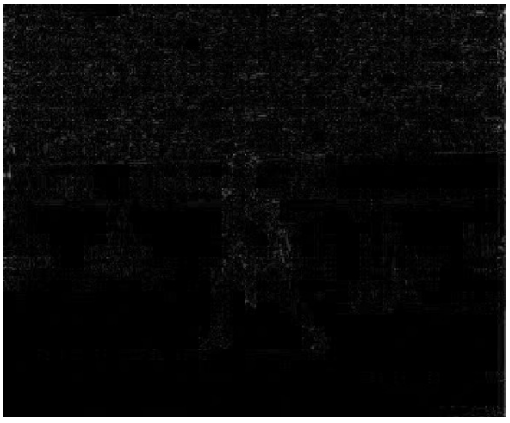
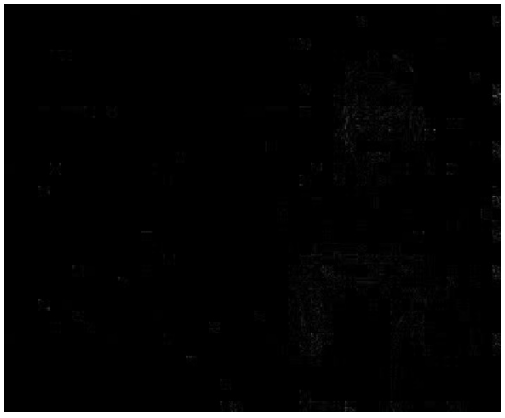
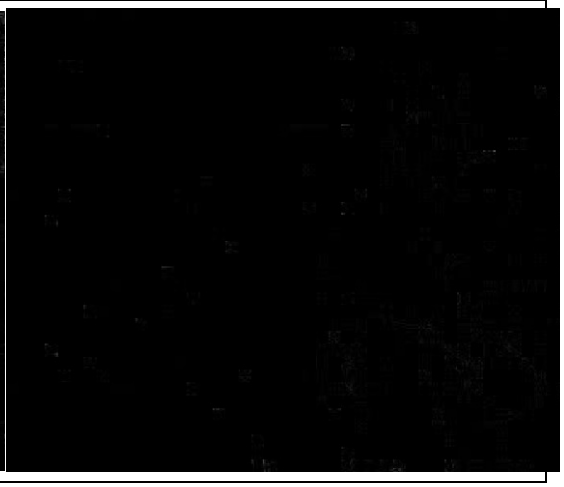
W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

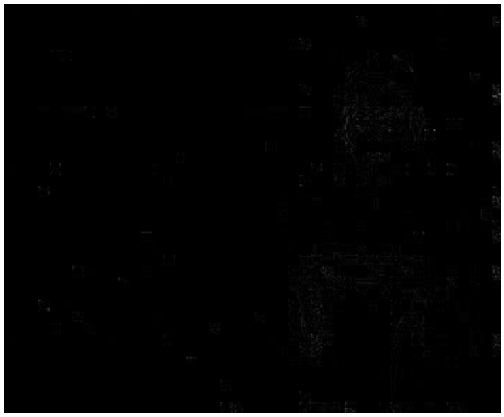
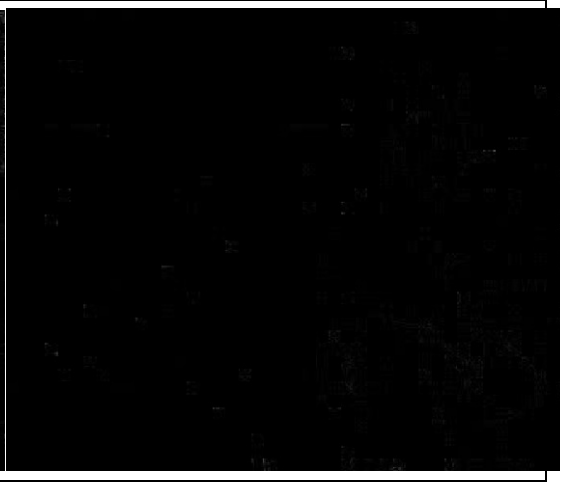
Fast W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

Fast W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

2. absolute residues sequences

W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

Fast W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

Fast W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

3. reconstructed sequences

W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

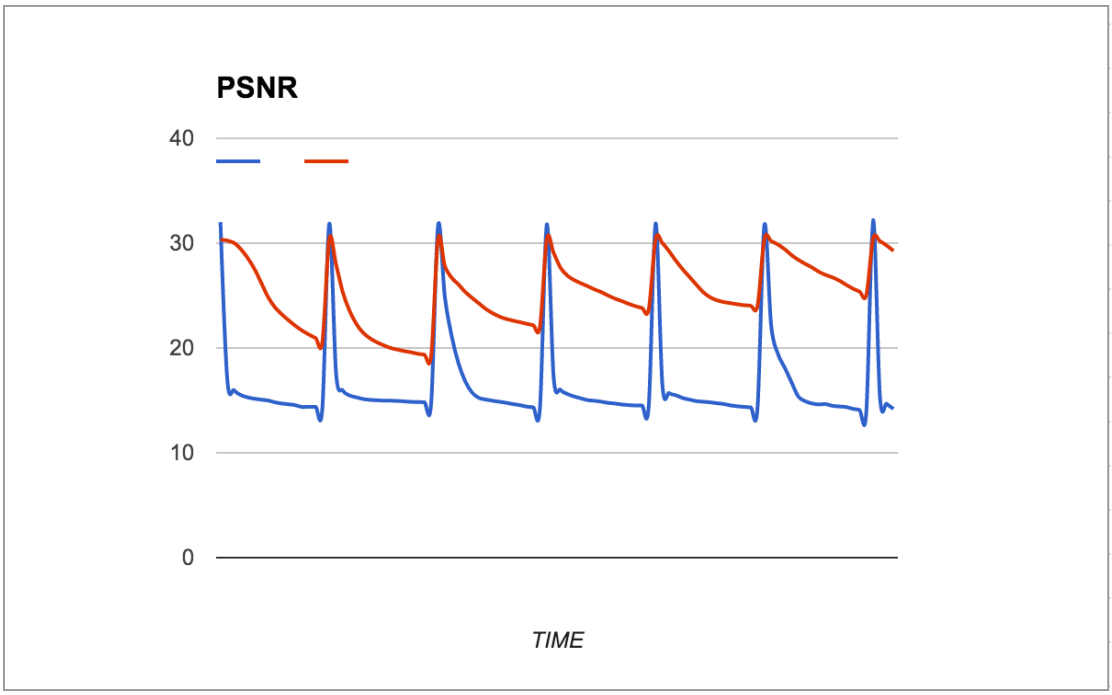
W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

Fast W8	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

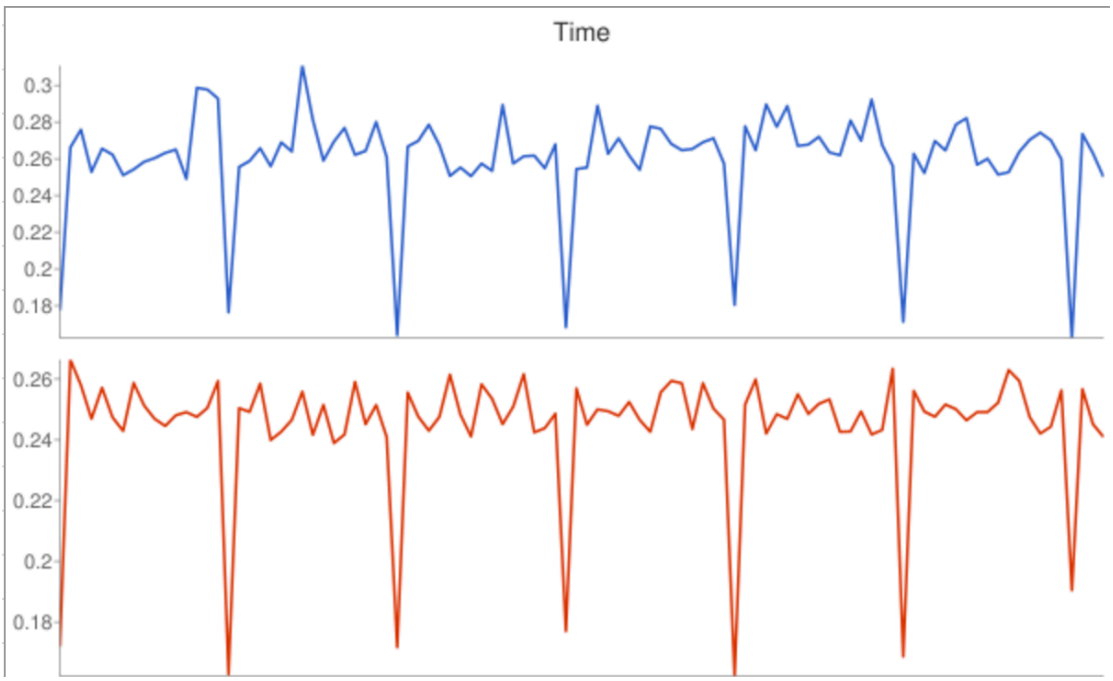
Fast W16	
Stefan 77	Weather 77
	
Stefan 88	Weather 88
	

4. PSNR & Time

由於各個 Search Range Mode 的 PSNR 與 Time 變化率差不多因此僅放 W8 來做說明，藍色線代表 stefan 而紅色代表 weather。



PSNR 部分可以發現每（GOP = 16）16 個 Frame 的 PSNR 會衝很高，因為那是 INTRA Frame，而往後的 INTER Frame 則會慢慢衰減直到 INTRA Frame。



Time 的部分會發現可以發現每 16 個 Frame 的 Time 會很低，因為那是 INTRA Frame 可以用較快的時間找到 MV，其變化與原圖差不多，而 INTER Frame 的時間則是跳動的。

stefan-w8	84.55231s
stefan-w16	261.349667s
stefan-w8-fast	26.523008s
stefan-w16-fast	26.040168s
weather-w8	89.788135s
weather-w16	273.534383s
weather-w8-fast	25.629405s
weather-w16-fast	24.453101s

由執行時間可以發現，Search Range 越大時間越久，但是使用 Three Step 實則差異不大。

5. Conclusion

i. 首先可以發現 Search Range 越大重建的效果越好，因為其預測準確率提升，但如 stefan 因為背景觀眾複雜不易找到 MV 因此其效果較差。

ii. Three Step 重建快速移動的物體與 Full 重建的結果差異不大。

iii. Three Step 重建之 PSNR 與 Full 差不多（由程式產生）。

iv. 此次作業結合 HW2，的部分並沒有用到 HW3 的 RLC RLD VLC VLD 部分，因此直接移植使用並沿用 HW3 的 Class 保留 DCT IDCT Quantization IQuantization，由這四個作業中學到了 Video Coding Base Line 技術，對於往後的相關研究定能更容易上手。