EE 392J Final Project

Phase-Correlation Motion Estimation

Yi Liang
yiliang@stanford.edu

Abstract

Phase-correlation motion estimation is studied and implemented in this work, with its performance, efficiency and complexity compared to that of the block-matching method. Since the phase-correlation method measures the motion directly from the phase correlation map, it gives a more accurate and robust estimate of the motion vector, and a motion field with much lower entropy. Phase-correlation method is very computationally efficient and performs better than the block-matching method having the same complexity does in most cases, especially on translational and large-scale motion.

I. Introduction

Interframe processing is the key to exploit and reduce the temporal redundancy in digital video compression. Temporal redundancy exists due to the similarity between the sequential neighboring frames [1]. In video compression, knowledge of motion helps to exploit this similarity and remove the temporal redundancy between neighboring frames in addition to the spatial and spectral redundancies [2]. Motion estimation (ME) and compensation (MC) are the basic approaches to find out and represent the motion between frames. These techniques are widely used in video standards including H.26x and MPEG to achieve high data compression rate.

In this work, we mainly study the phase-correlation motion estimation method, which is block-based, and compare its performance, efficiency and complexity to that of the conventional block-matching (BM) method. Block-matching motion estimation and compensation are popular approaches in practice due to their robust performance and no need for object identification. However the complexity and computation time for BM might be high in order to obtain good performance since it searches the target frame to find the matching block. The proposed phase-correlation ME, in contrast, has much lower complexity by measuring the motion directly from phase correlation, and gives much smoother motion vector field. Since phase-correlation is also block-based, its performance can be fairly compared with that of the conventional block-matching method in many ways.

II. Block-Based Motion Estimation Methods

Common motion estimation algorithms being used include block-based methods, optical flow methods (OFE), and pel-recursive methods. Among these, block-based methods are of current interests due to their low overhead to represent the motion vector field (MVF) and the availability of hardware implementations [1].

In a typical block-based method, the current frame is divided into blocks and the purpose of ME is to find out the corresponding motion vector for each block according to its relative displacement from the previous frame. There are several block-based motion estimation methods. One of them popularly employed in video standards is the block matching (BM) method. In BM, the best motion vector estimate is found by a spatial-domain search procedure. The object block of the current frame is placed and moved around in the previous frame using a specific search strategy. A criterion is defined to determine how well the object block matches a corresponding block in the previous frame. These criteria can be mean squared error (MSE), minimum absolute difference (MAD) and sum absolute difference (SAD). Since MSE gives the residual energy in the block difference, it is used as the criteria for BM ME in later sections in order to make comparisons between BM and phase-correlation algorithms. Search procedures for BM ME include full search, logarithm search, and hierarchical method. We will use full search and logarithm search for later comparisons since they are based on fixed block size. Phase-correlation is also one of the block-based methods, whose detail is given in the next section.

III. Phase-Correlation Method

1. Phase Correlation and Motion Estimation Implementations

Unlike the BM method, which searches the blocks from luminance matches, the phase-correlation method measures the movement between the two fields directly from their phases. The basic principles are briefed as below.

Assuming a translational shift between the two frames

$$s(n1,n2,k)=s(n1+d1,n2+d2,k+1) \qquad \text{Eq. 1}$$

Their 2-D Fourier transforms are

$$S_k(f1,f2)=S_{k+1}(f1,f2)\exp[j2\pi(d1f1+d2f2)] \qquad \text{Eq. 2}$$

Therefore the shift in the spatial-domain is reflected as a phase change in the spectrum domain. The cross-correlation between the two frames is

$$c_{k,k+1}(n1,n2)=s(n1,n2,k+1)**s(-n1,-n2,k) \qquad \text{Eq. 3}$$

whose Fourier transform is

$$C_{k,k+1}(f1,f2)=S_{k+1}(f1,f2)S_k*(f1,f2) \qquad \text{Eq. 4}$$

In order to get rid of the luminance variation influence during our phase analysis, we normalize the cross-power spectrum by its magnitude, and obtain its phase

$$\Phi[C_{k,k+1}(f1,f2)] = \frac{S_{k+1}(f1,f2)S_k*(f1,f2)}{\left|S_{k+1}(f1,f2)S_k*(f1,f2)\right|} \qquad \text{Eq. 5}$$

By Equation 2 and 5, we have

$$\Phi[C_{k,k+1}(f1,f2)] = \exp[-j2p(f1d1+f2d2)] \qquad \text{Eq. 6}$$

whose 2-D inverse transform is given by

$$c_{k,k+1}(n1,n2)=\delta(n1-d1,n2-d2) \qquad \text{Eq. 7}$$

As a result, by finding the location of the pulse in Eq. 7 we are able to tell the displacement, which is the motion vector. In practice, the motion is not pure translational, we will get the phase correlation similar to what is depicted in Fig. 1. In this case we locate the pulse by finding the highest peak or a few candidates.

In implementation, the current frame is divided into 16x16 blocks and phase correlation calculation is performed for each block. In order to correctly estimate the cross correlation of

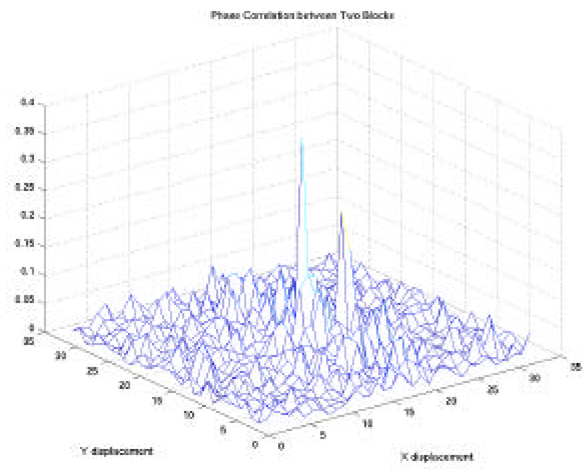
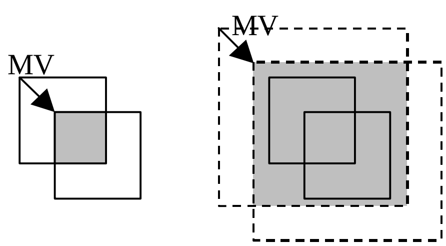


Fig. 1 Phase Correlation between Two Blocks

corresponding blocks in respective frames, we extend the blocks to 32x32 in size, centered around the formerly defined 16x16 blocks, to calculate phase correlation. If we do this only for 16x16 blocks, their correlation might be very low for particular motion due to the small overlapping area, as shown in Fig. 2(a). Once the block size is extended to 32x32, the overlapping area is increased for better correlation estimation, as is shown in Fig. 2(b). Obviously there will be overlap among the extended blocks and a pixel usually exists in multiple object blocks.

A two-dimensional raised-cosine window is applied to each 32x32 object block to give more weight to our formerly defined 16x16 region, to which a motion vector will be assigned. Experiments show that in most cases (see Table 2 and Table 3) the weighting window helps to reduce the energy in the displacement field difference (DFD) by 0.1-0.2dB while the added complexity (reflected by computation time) is almost negligible.



(a) (b)

Fig. 2 Correlation Area Using 16x16 (a) and 32x32 (b) Blocks

To calculate the phase correlation of the blocks in respective frames, 32x32 FFTs are first performed on the two corresponding blocks. And the inverse FFT of the multiplication of one spectrum and the conjugate of the other is their phase correlation. In practice, the highest peak (or several candidates) is picked out from the phase correlation matrix to estimate the displacement vector, as shown in Fig.1.

A maximum displacement of +/-16 is available due to the size of the FFT and inverse FFT. In some cases, we may want to restrict the maximum shift within a certain range by cropping the center part of the phase correlation matrix. In our examples, we already had some idea that the motion vector components can be large, there for we allow the maximum shift to be +/-16.

2. Image Correlation

Although the phase correlation peak gives us some idea on the displacement between the blocks, it doesn't tell us whereabouts within the block the movement takes place. For example, in the two examples (a) and (b) in Fig. 3, although the locations of moving objects are different, they result in the same correlation map.

The highest peak in the correlation map usually corresponds to the best match between a large area, while not necessarily the best match for our 16x16 object block. If there are several moving objects in the block with different displacement, there can be several peaks appearing in the correlation map, as shown in Fig. 1, where we see two peaks. In this case we may want to select several candidates instead of just one highest peak, and then decide which peak best represents the displacement vector for the object block. Once the candidates are selected, we examine them one by one using image correlation. For each candidate, the motion vector is already found, hence the 16x16 object block can be placed in the 32x32 window of the previous frame to measure the extent of correlation. The candidate resulting in the highest image correlation is the one we want, and its displacement is the right motion vector for the object block.

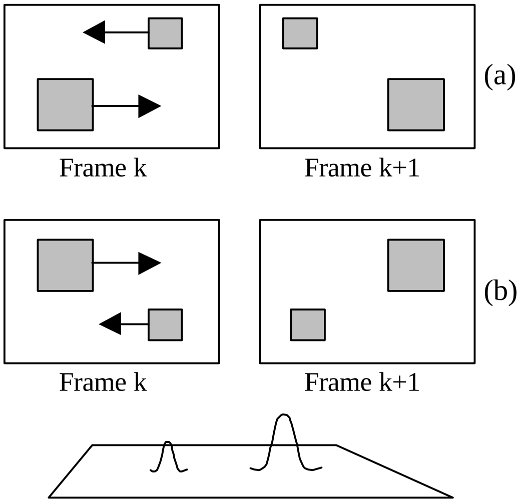


Fig. 3 Two Examples Giving the Same Correlation Map

Image correlation is in fact a matching procedure, which is similar to the BM method, except that image correlation is performed after the displacement vectors are already found. Therefore the computation time is greatly reduced by not trying to search the whole area.

3. Half-Pixel Motion Estimation

Although digital video is represented by pixels, the motion in not necessarily limited to integer number of pixel offset. Representing fractional motion vector gives sub-pixel accuracy to motion compensation. Half-pixel improves the ME performance since it also reduces noise as it averages and interpolates the pixels.

In the phase-correlation method, we estimate possible half-pixel offset from the correlation map after the integer motion vector has been found. We interpolate the correlation at half-pixel offsets by examining three neighboring correlation samples adjacent to the found peak using cubic spline interpolation. If the interpolated correlation supercedes the previously found highest integer value, the motion vector is updated to the new offset, which is not an integer. Fig. 4 depicts this procedure for 1-D case. Samples represented by 'x' are interpolated from their neighboring samples with integer offsets, which are represented by 'o'. The previously found offset is zero, but the correlation interpolated at 0.5 is even higher now. As a result, displacement zero is updated to the non-integer value 0.5.

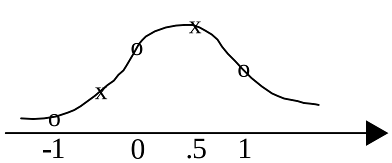


Fig. 4 Interpolation of Correlation Map

In order to make comparisons between BM and phase-correlation, half-pixel ME is also applied in BM method. In BM method, the best integer MV is found first. The selected block in the previous frame is then interpolated and compared to the object block. The same criteria, such as MSE in our experiments, is used to search for the best match. The MV will be updated to the new found offset which minimizes the matching criteria if it is not integer.

Experiments show that half-pixel ME greatly improves the performance. In phase-correlation method, the PSNR of DFD is improved by an average of 0.5dB for most cases (Table 1 and Table 2), while for BM, the improvement is about 0.6dB for most cases.

IV. Comparison Results

In this section we examine a few examples and compare the performance, efficiency, and complexity of BM and phase-correlation methods for motion estimation. For either method, the current frame is divided into 16x16 blocks with each block having a motion vector calculated. Hence we obtain the same number of motion vectors for a particular frame from either algorithm. The maximum offset allowed is +/-16 pixels. The frames are luminance signal retrieved from SIF sequences bus, calendar and carousel, and QCIF sequence car phone.

For each example, the frame difference is measured in PSNR first. The PSNR resulting from full search BM method is also listed. Given a certain range of allowed offset, the full search BM always offers the best performance since it finds the matching block through a thorough search procedure. But the full search is far more computationally intensive than other methods (Table 1). For these reasons, we list the full search BM PSNR as a reference only. The comparison is carried out only between the four-step log-search BM and the phase-correlation method. These two methods have the complexity on the same order.

In order to compare with the phase-correlation method using weighting window, the same raised-cosine weighing function and overlapping windows are also applied in BM method. However the performance turns out worse compared to conventional BM due to the extension of the smaller block (Table 1). Hence we stick to using BM without weighting window for comparisons.

In log search BM, the criterion applied is MSE to better demonstrate the energy in the DFD. For the phase-correlation with image correlation, two highest peaks are selected and image correlation is calculated for both of them to search for the best match.

1. Comparison Criteria

We use the following criteria for our comparisons.

i) Peak signal-to-noise ratio (PSNR) of the DFD. The PSNR tells us how much residual energy is in the DFD and hence the compression efficiency.

ii) Entropy of the motion field. This is a measure of the smoothness of the motion field and how many bits should be spent on the motion during transmission. Lower entropy means cheaper transmission.

iii) Complexity of the algorithm. This is measured by the computation time. All the computations are performed on Intel Pentium III machines with Linux OS, which give impressive performance on integer executions.

iv) Subjective opinion on the motion compensated image.

2. Results

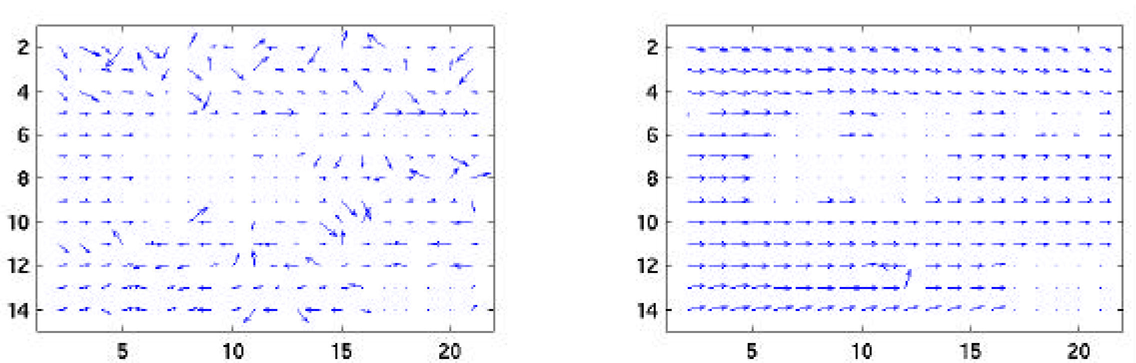


Fig. 5 Motion Field by BM (left) and Phase Correlation (right), Bus Frames

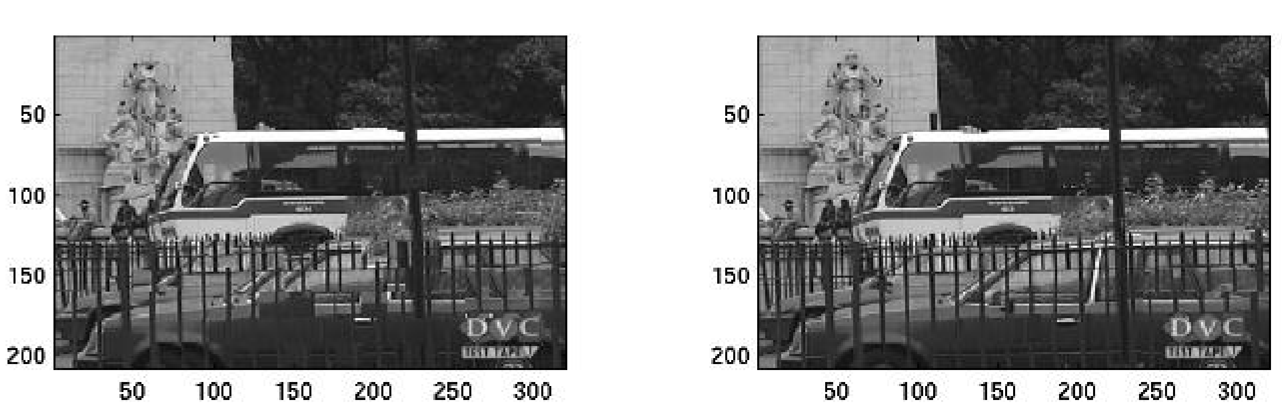


Fig. 6 Prediction by BM (left) and Phase Correlation (right), Bus Frames

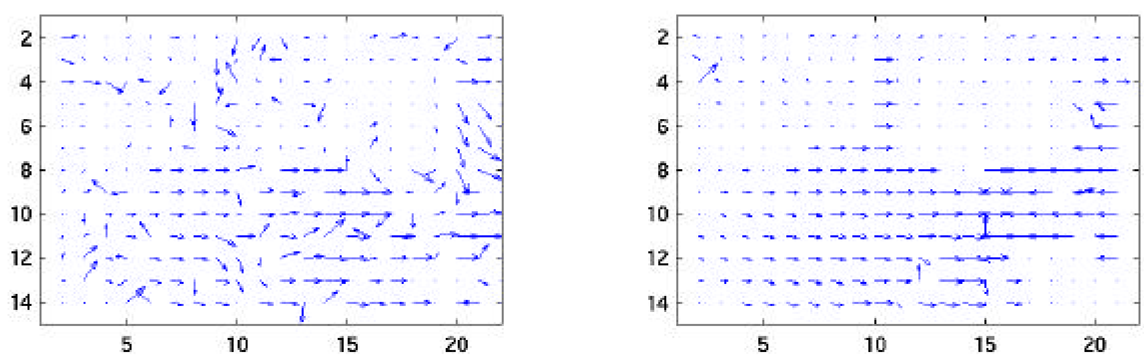


Fig. 7 Motion Field by BM (left) and Phase Correlation (right), Carousel Frames

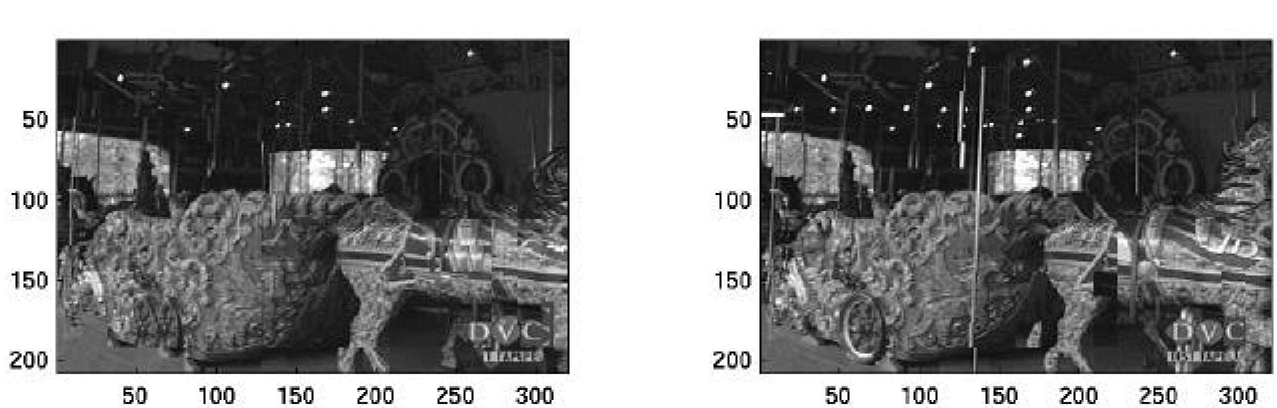


Fig.8 Prediction by BM (left) and Phase Correlation (right), Bus Frames

	PSNR of DFD (dB)	Entropy of Motion Vector Field (bits)	ME Computation Time (sec)
Frame Difference	16.06		
Full Search BM	24.14	4.31	57.75
Full Search BM, with Weighting Window	22.93	5.24	63.25
Log Search BM	21.06	6.63	1.82
Log Search BM with Weighting Window	19.72	7.09	2.64
Log Search BM with Half-pixel ME	21.67	7.87	2.00
Phase Correlation	22.72	2.15	1.47
Phase Correlation with Image Correlation	22.78	2.08	1.71
Phase Correlation with Weighting Window	22.66	2.15	1.73
Phase Correlation with Half-pixel ME	23.37	2.67	2.85

Table 1 Performance Comparison of Different Algorithms, Bus Frames

	PSNR of DFD (dB)	Entropy of Motion Vector Field (bits)	ME Computation Time (sec)
Frame Difference	12.67		
Full Search BM	20.55		
Log Search BM	17.24	7.87	1.82
Log Search BM with Half-pixel ME	17.94	9.37	1.98
Phase Correlation	19.71	5.00	1.52
Phase Correlation with Image Correlation	19.38	5.08	1.72
Phase Correlation with Weighting Window	19.89	4.95	1.47
Phase Correlation with Half-pixel ME	20.51	6.02	2.86

Table 2 Performance Comparison of Different Algorithms, Calendar Frames

	PSNR of DFD (dB)	Entropy of Motion Vector Field (bits)	ME Computation Time (sec)
Frame Difference	15.52		
Full Search BM	21.41		
Log Search BM	19.48	7.50	1.82
Log Search BM with Half-pixel ME	19.91	8.83	1.98

Phase Correlation	17.82	4.97	1.47
Phase Correlation with Image Correlation	17.72	6.14	1.72
Phase Correlation with Weighting Window	17.90	6.45	1.47
Phase Correlation with Half-pixel ME	18.03	6.80	2.83

Table 3 Performance Comparison of Different Algorithms, Carousel Frames

	PSNR of DFD (dB)	Entropy of Motion Vector Field (bits)	ME Computation Time (sec)
Frame Difference	28.40		
Log Search BM	31.95	2.36	0.45
Log Search BM with Half-pixel ME	32.67	3.90	0.49
Phase Correlation with Weighting Window	29.60	0.83	0.37
Phase Correlation with Half-pixel ME	30.25	1.41	0.75

Table 4 Performance Comparison of Different Algorithms, Car Phone Frames (QCIF)

For the bus frames, we have tried all different algorithms of BM and phase correlation. For the other examples, we mainly listed results by four-step log search BM, phase-correlation with and without weighting window, and half-pixel ME.

From the calculated motion field entropy of all the examples and the motion vector field plots we can see that the motion field given by phase correlation is much smoother with much lower entropy. The phase-correlation method performs better in this aspect because the log search is easier to give erroneous motion estimation due to the limited number of searches it carries out. In comparison phase-correlation measures the displacement directly instead of blindly searching for it.

In terms of complexity, phase correlation is about 20% faster if not using half-pixel ME. That's because fast algorithms exist for FFT. The BM is faster if half-pixel ME is used. This partly results from different interpolation schemes used for BM and phase-correlation.

In terms of prediction frame quality, from bus and calendar examples we observe better prediction by phase-correlation. We also observe that the phase-correlation method produces higher DFD PSNR of more than 1dB. With the complexity on same order, phase-correlation is able to measure the motion vector more accurately and is more robust in general. This is especially true in the case if motion is largely translational.

The carousel example is an exception to the above, where the phase-correlation produces poorer PSNR since the frames involve rotational and camera zoom in/out motion and multiple-object movement. However the prediction by phase-correlation looks better by subjective opinion and preserves the original frame features better. For example the bright pole in the middle is well predicted while the BM didn't make it (Fig. 8).

For both methods, half-pixel ME makes a big difference for the reasons stated in section III-3.

The two ME methods are also applied to QCIF frames with less motion. The BM works better than phase-correlation by around 2dB in DFD PSNR (Table 4). The reason is that in the QCIF frames we have, the motion is tiny, infrequent and takes place in small regions. Phase-correlation is less capable of capturing motion in a small region since it has to extend the block size and prefers larger coverage areas. In the SIF frames we showed above, the motion tends to be in large scale.

V. Conclusions

Phase-correlation ME is very computationally efficient and it produces much smoother motion field with low entropy than the BM method does. Phase-correlation works better than the BM does in the cases of large scale translational motion, while BM is more suitable for predicting regular and small scale motion and multiple-object movement.

It is also found that the block size greatly affects the ME performance. The blocks should be large enough to group pixels with similar motion but should be small enough to separate pixels with different motion and multiple-object movement.

References

[1] A. M. Tekalp, *Digital Video Processing*, Prentice Hall, 1995.

[2] C. Stiller, J. Konrad, "Estimating Motion in Image Sequences", *IEEE Signal Processing Magazine*, pp.70-91, July 1999.

[3] J. Watkinson, *The Engineer's Guide to Motion Compensation*, Snell & Wilcox, 1994