LIMITED MOTION ESTIMATION SCHEME FOR MULTIMEDIA VIDEO COMPRESSION

A.N. Evans, Y. Guo and D. M. Monro

Department of Electronic and Electrical Engineering, University of Bath, Claverton Down, Bath, BA2 7AY, UK. Email: A.N.Evans@bath.ac.uk, D.M. Monro@bath.ac.uk

ABSTRACT

This paper presents a computationally efficient motion estimation technique based on image sampling which determines the dominant motion between pairs of images. The technique is suited to low complexity, low bit rate multimedia applications, where the objective is to achieve good fidelity without the overhead of full motion compensation. This can be achieved if the dominant motion is a combination of translation, rotation and zoom, which can be described by a similarity transformation. The method adopts a new approach to determining the model parameters, based on generating a list of parameter estimates from pairs of block motion vectors and selecting the mean of those estimates close to the median. The method gives a good sub-pixel dominant motion estimate by sampling as little as 1/20th of the image area. Results show the method to be accurate and robust, with low computational requirements.

1. INTRODUCTION

Motion estimation is an important component of video codecs, as it greatly reduces the inherent spatial redundancy within video sequences. However, it also accounts for a large proportion of the computational effort. To estimate the motion of pixels between pairs of images block matching algorithms (BMA) are regularly used, a typical example being the Exhaustive Search Algorithm (ESA) often employed by MPEG-II. Many researchers have proposed and developed algorithms to achieve better accuracy, efficiency and robustness [1-5]. A common approach is to search in a coarse to fine pattern or to employ decimation techniques. However, the saving in computation is often at the expense of accuracy. This problem has been overcome by the successive elimination algorithm (SEA) of Li and Salari [6], that produces identical results to the ESA with greatly reduced computation, and is the method used in this research. However, block-based motion estimation still remains a significant computational expense and is sensitive to noise. A further disadvantage of a block-based approach is that the motion vectors constitute a significant proportion of the bandwidth, particularly at low bit rates. This is one reason why standard systems such as MPEG II or H263 use larger block sizes.

In typical multimedia video sequences, many image blocks share a common motion, as scenes are often of low complexity. If more than half the pixels in a frame can be regarded as belonging to one object, we define the motion of this object as the dominant motion. This definition places no further restrictions on the dominant object type; it can be a large foreground object, the image background, or even fragmented. A model of the dominant motion represents an efficient motion coding scheme for low complexity applications such as those found in multimedia and has become a focus for research during recent years [7-9]. For internet video broadcast, a limited motion compensation scheme of this type offers a fidelity enhancement without the overhead of full motion estimation.

The use of a motion model can lead to more accurate computation of motion fields [10] and reduces the problem of motion estimation to that of determining the model parameters. One of the attractions of this approach for video codec applications is that the model parameters use a very small bandwidth compared with that of a full block-based motion field.

The paper is organized as follows. The motion model and new algorithm are described in Section 2 and in Section 3 the optimal block size for the new method is determined. Section 4 presents experimental results and conclusions are drawn in Section 5.

2. MOTION MODEL AND ALGORITHM DESCRIPTION

For many multimedia applications, the dominant motion can be described by a similarity transform that has only 4 parameters. As shearing is relatively rare in most video sequences its exclusion does not compromise the generality of the model. The similarity model relates corresponding points in the source (x,y) and object (u,v) images by

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix} \quad (1)$$

For motion estimation the parameters a, b, c and d are unknown and must be inferred from image pairs using correspondent points and a matrix inverse operation. When errors in the matching process are anticipated, an over-determined set of equations can be

0-7803-5682-9/99/$10.00©1999 IEEE.

used to approximate the parameters. The conventional approach is to use a least-squares technique, perhaps with regularization, to solve the matrix inverse. However, for real-time motion estimation applications this has the disadvantages of being computationally expensive and sensitive to outliers.

A new method for estimating the model parameters is presented that, instead of combining all matching blocks in one least-squares procedure, takes them in pairs to produce a series of estimates for the parameters. Equation (1) has four unknowns and thus can be solved using two image points, *{(x1,y1), (x2,y2)}* and their correspondences *{(u1,v1), (u2,v2)}*. Substituting these into (1) and eliminating c and d gives

$$\begin{bmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (y_2 - y_1) & (x_1 - x_2) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} u_2 - u_1 \\ v_2 - v_1 \end{bmatrix} \quad (2)$$

which can be easily solved either directly or, for example, by Gaussian elimination. Thus each pair of matched points produces an estimate of a and b, but with a high risk of error. The problem now is to determine accurate values from the list of estimates.

As the dominant motion is the motion of the majority of the blocks, many of the estimates should be of similar parameter values. Therefore the most common value of the estimates should be that of the dominant motion. This suggests that a histogram-type approach can be used, with the dominant motion parameters corresponding to the mode. However, selecting the accumulator bin size for a histogram is not trivial, a problem exacerbated when the number of estimates is small compared with the numbers of bins and a high resolution estimate is required. This has motivated a new approach to parameter estimation, using Order Statistics.

Figure 1 shows a typical sorted list of estimates from an actual image pair for the parameter a. For this example 20% of the 1024 blocks in a test image were sampled, giving 102 estimates. Presented in this format, the modal value occurs at the flattest section of the graph, as this is produced by a significant number of estimates with very similar values. The flat region is approximately 24 estimates wide. Those estimates on either side of the flat region result from incorrect or inaccurate motion estimates, and from blocks not belonging to the dominant object. It should be noted that the distribution of these outliers is reasonably symmetric around the central flat region. The mean of the estimates within the flat region offer a suitable value for the parameter a and this is implemented in practice by averaging those estimates within ±0.1 standard deviations of the median value. This approach is used to find values for a and b, which are substituted into (1) to produce a list of estimates for c and d, from which values can be found using the same method. The method is easily combined with a block sampling scheme and a pseudo-algorithm is given in Figure 2.

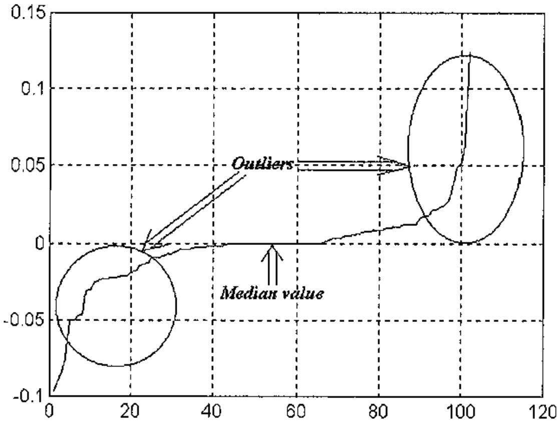


Fig. 1: Ranked list of estimates for the parameter a

```
Step 1 Tile the reference frame into non-overlapping blocks.
While (Proportion of blocks selected < desired proportion)
    Step 2 Select blocks from the source image and use SEA algorithm to find matching blocks
    Step 3 Find estimates for parameters $a$ and $b$ using Equation (2) and place them in a sorted lists.
End
Step 4 The mean of the estimates within ±0.1 standard deviations of the median provides values for $a$ and $b$.
Step 5 For each block, substitute the calculated values for $a$ and $b$ into Equation (1) and solve to give estimates for the translational motion components $c$ and $d$. Again, place the estimates in a sorted list.
Step 6 Find the mean of the estimates within ±0.1 standard deviations for $c$ and $d$.
```

Fig. 2: Algorithm for model parameter estimation.

3. BLOCK SIZE SELECTION

Step 2 of the algorithm described in Figure 2 uses block comparisons to find matching blocks. This is performed using the SEA algorithm and the mean absolute deviation (MAD) is the matching metric. However, the question of block size still has to be addressed. It can be seen from Figure 1 that there must be sufficient values in the flat region of the list for the parameter selection routine to be successful. The number of blocks that must be sampled to achieve this is proportional to the probability of finding pairs of good matches and the size of the dominant object.

Therefore the smaller the block size, the less the computation that is required to generate the list of estimates. Furthermore, small blocks have less likelihood of containing points from more than one object. Opposing this, the probability of achieving a good match decreases with smaller block sizes.

To investigate this effect a test set of image pairs with known inter-image transformation parameters were generated from the Y component of the CCITT test

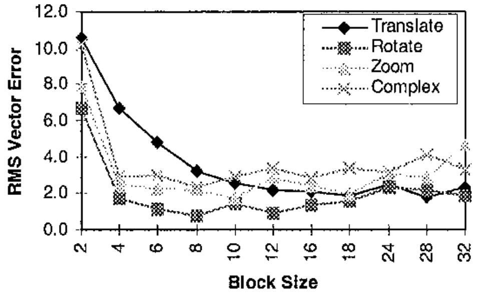


Fig. 3: RMS vector errors of raw motion vector estimates versus block size.

image Gold Hill. The test set consisted of pure translation, rotation and zoom and a combination of the three, termed complex motion, that consisted of a combination the pure motions: 4.5 pixel horizontal and vertical shift, a zoom of 1.035 and finally a 3° rotation.

The reference coordinate for the transformation was the image center and the image size was 256x256 for all cases. For each test image pair the SEA was repeatedly applied, varying the block size between 2 and 32 pixels. The search range was set to ±8 pixels for the pure motion test images and ±16 pixels for the complex motion pair. Figure 3 presents the RMS vector error between the raw motion estimates from matching the all blocks in the test images, using a range of block sizes between 2x2 and 32x32 pixels. It can be seen that the overall error for all test images is at a lowest within the range of 8 to 16 pixels and therefore the minimum of this (8x8) was selected for the block size.

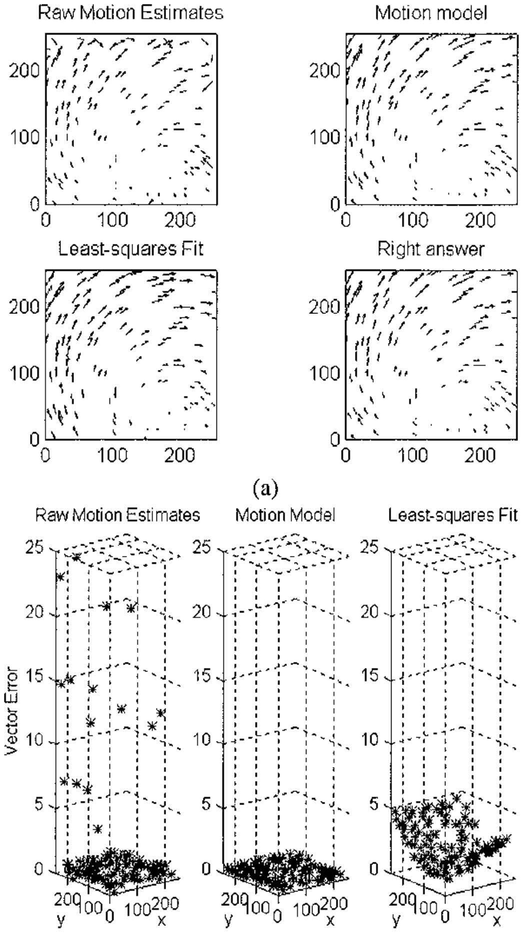
4. EXPERIMENTAL RESULTS

The new algorithm was applied to the test image set. In all cases 20% of the image blocks were sampled, the block size was 8x8 and the search range was as above. To quantify the performance of the algorithm, the results achieved are compared with those of the raw motion estimates and those produced by performing a least-squares fit on the raw estimates. In addition, the known inter-frame transformation parameters are used to provide a set of standard results (the "right answer") which can be directly compared with the model results.

Figure 4 (a) presents the motion vectors produced for the complex motion test image set. The raw block motion estimates (top left) are irregular and inconsistent. Both the motion model and least-squares fit exhibit a smooth field but by comparison with the right answer (bottom right) it can be seen that the motion model result is closer to that of the actual flow.

This is confirmed by considering the vector error at each point. The vector error is the difference between the detected and the true motion vectors, given by

$$\sqrt{\left(dx_1 - dx_2\right)^2 + \left(dy_1 - dy_2\right)^2} \qquad (3)$$



(a)

(b)

Fig. 4: Experiment results for complex motion (a) motion fields and (b) vector errors

and shown in Figure 4(b). The errors for the motion model and least-squares fit are much lower that of the block matching estimates. However, the sensitivity of the least-squares technique to outliers can be seen in the residual error values. The motion model techniques clearly has the lowest overall error and this is confirmed by Figure 5 which gives the average vector error for pure translation, rotation, zoom, and the complex motion of Figure 4(b). For all cases the motion model has produced the lowest error, with values ranging from a half to a sixth of the least-squares result.

The performance of the motion method in the

	Raw Estimates	Motion Model	Least-Squares Fit
Translation	1.98	0.61	1.22
Zoom	1.63	0.24	1.51
Rotation	0.94	0.29	0.71
Complex	2.49	0.33	2.43

Table 1: Average RMS vector errors for all test images

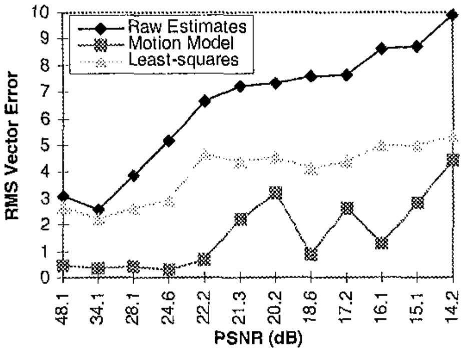


Fig. 5: RMS vector error in presence of noise for complex motion test image set

presence of noise is assessed by adding Gaussian noise with zero mean and standard deviation 1.0 to the test image set, to a predetermined level of Power Signal to Noise Ratio (PSNR). The result for complex motion is shown in Figure 5. Again the motion model produces the lowest error; this was also the case for the pure translation, rotation and zoom test images.

Finally the performance in relation to the proportion of the image of the image that is sampled is investigated. Figure 6 shows the results for the complex motion test images. Between 16 and 200 of the 1024 8x8 blocks were sampled. Below 16 the vector error rapidly increased. For all cases the model method produces a lower error than the other techniques. In practice, high quality results can safely be achieved with as little as 48 blocks, less than 5% of the image.

5. CONCLUSIONS

A new low complexity limited motion estimation algorithm has been described, based on image sampling. The underlying model for our algorithm is the similarity transform, which only requires 4 model parameters to specify the flow vectors and is well suited to multimedia applications.

The conventional approach to estimating the parameter values uses motion vectors and a least-squares technique. This is both computationally expensive and sensitive to outliers. Instead of combining all vectors in one estimate, our method generates a list of estimates from pairs of matching blocks. The mean of those estimates close to the median is then selected as the parameter value. This is a robust nonlinear technique that produces accurate results with reduced computation. The approach can be used for many practical multimedia applications and has the potential for extension to multiple objects. We have evaluated the method and found it to be accurate and efficient for determining the dominant motion using as little as 5% of the image area

Our group have recently developed a two layer video codec that demonstrated 10dB improvement over so-called Motion JPEG [11] for fixed cameras and no motion compensation. The technique presented here will further improve this system, with significantly lower computation than occurs in block based motion compensated video systems such as MPEG-II or H263. We also plan to extend the technique to allow the extraction of multiple video objects.

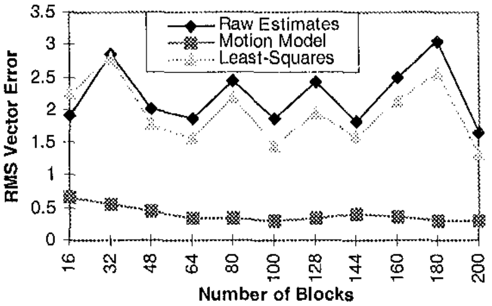


Fig. 6: RMS vector error versus number of blocks sampled for complex motion test images

6. REFERENCES

[1] T. Koga, K. Iinuma, A. Hirano, Y. Iijima and T. Ishiguro, "Motion-compensated interframe coding for video conferencing", *National Telecommunications Conf.*, pp. G 3.1-3.5, 1981.

[2] R. Srinivasan and K.R. Rao, "Predictive coding based on efficient motion estimation", *IEEE Trans. Communications*, vol. 33, pp.888-896, 1985.

[3] S. Kappagantula and K.R. Rao, "Motion compensated interframe image prediction", *IEEE Trans. Communications*, vol. 29, pp.1011-1015, 1985.

[4] B. Liu and A. Zaccarin "New fast algorithms for the estimation of block motion vectors", *IEEE Trans. Circuits Systems Video Technol.*, vol. 3, no. 2, pp. 148-157, 1993.

[5] Lee X., and Zhang Y.Q. "A fast hierarchical motion-compensation scheme for video coding using block feature matching", *IEEE Trans. Circuits Systems Video Technol.*, vol. 6, no. 6, pp. 627-635 1996.

[6] W. Li and E. Salari, "Successive elimination algorithm for Motion Estimation", *IEEE Trans. Image Processing.*, vol. 4, no. 1, pp. 105-107, 1995.

[7] S. Pei C. Ko and M. Su, "Global motion estimation in model-based image coding by tracking three-dimensional contour feature points", *IEEE Trans. Circuits Systems Video Technol.*, vol. 4, pp. 257-275, June 1994

[8] T. Sikora, "MPEG digital video-coding standards", *IEEE Signal Processing Magazine*, pp. 82-100, 1997.

[9] M.C. Lee, W. Chen, C.B. Lin, C. Gu, T. Markoc, S.I. Zabinsky and R. Szeliski, "A layered video object coding system using sprite and affine motion model", *IEEE Trans. Circuits Systems Video Technol.*, vol. 7, no. 1, pp. 130-145, 1997.

[10] M.I. Sezan and R.L. Lagendijk. *Motion Analysis and Image Sequence Processing. Kluwer Academic Publishing*, pp. 1-25, 1993

[11] D.M. Monro, H. Li and J.A. Nicholls, "Object based video with progressive foreground", IEEE Int. Conf. Imag Process. (ICIP97), Vol III, pp. 48-52, Santa Barbara Calif., October 1977