Adaptive subregion variable shape block motion compensated prediction

N W Garnham   M K Ibrahim

Department of Electrical and Electronic Engineering
University of Nottingham
University Park
Nottingham
NG7 2RD
United Kingdom

ABSTRACT

This paper describes a general approach to interframe video coding using a method of local feature-based motion compensation. Used in conjunction with a DPCM/DCT interframe coding technique, such as the CCITT H.261 algorithm, this technique is computationally viable at real-time rates, employing a simple process of evaluating the nature and content of features and their subsequent displacement. As a secondary means of interframe coding, it has proved possible to introduce mode-value blocking and quantisation to speed the motion estimation process and results indicate that as part of a hybrid encoding algorithm, this approach is quite acceptable as no long-term errors are propagated. In addition to a description of feature analysis as a combinational coding method, it will be shown that the process of feature extraction and evaluation provides, in its own right, scope to form the basis of a complete, efficient codec of low complexity.

Keywords: motion compensation, low bit-rate coding, video compression, video codecs, videoconferencing.

1. INTRODUCTION

Contemporary developments in the area of visual communications, broadcast video and multimedia have demonstrated the effectiveness of coding and compression algorithms that remove both spatial and temporal redundancy. Of all applications, however, the use of video coding in videoconferencing and videophony has shown the need for techniques that are at once efficient and cost-effective. The CCITT H.261 algorithm[1, 2] has formed the basis of developments in universal videoconferencing over channels of $m \times 64$ kbits/s, where using a technique known as Differential Pulse Coded Modulation (DPCM) and the Discrete Cosine Transform (DCT), spatial (intrafield) and temporal (interframe) redundancies are removed to produce a significant degree of video compression, whilst retaining a picture of acceptable quality.

However, whilst the process of pixel-based differencing is effective, in practice many videoconferencing scenarios contain displacements which are significant and introduce more difference values than the codec can process, given the trade-off between picture quality and the maintenance of a flow of data at a given rate. To this end, motion compensation is developed to augment the efficiency of the DPCM/DCT algorithm. Of the techniques available for motion compensation, block based methods have been pursued with most interest, allowing the use of low-intensity algorithms and simple displacement vectors. Jain and Jain[3] originally proposed a method involving the division of an image into small blocks of a given size. Using a larger search window, all possible blocks of the same size in the previous frame are evaluated and the position at which least errors occur is assumed as the origin of a current block and a displacement vector produced. The receiver uses this vector in conjunction with the transformed DPCM values to reconstruct the picture (figure 1). As a relatively simple algorithm, the block-matching technique has proven simple to implement and is adopted by the CCITT study group XV[4] in their base models for a videoconferencing codec system operating on channels of $m \times 64$ kbits/s and hardware solutions have been constructed[5, 6].

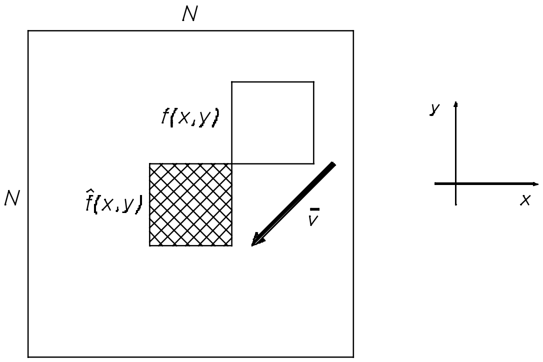
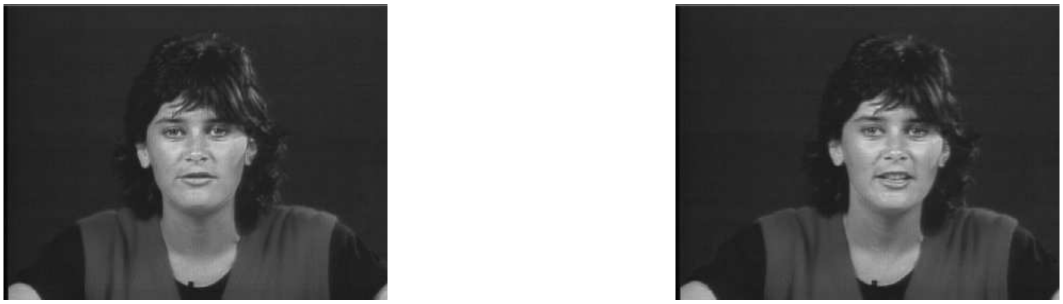


Figure 1: Block-matching motion compensation, showing the source of a current search block *f'(x,y),* within an $N \times N$ search window.

The fundamental drawback with the full-search block matching motion compensation techniques described is that whilst the algorithm is simple, it is computationally very intensive and work has shown that it is impractical for implementation in commercial low-cost video codecs. Furthermore, the algorithm assumes that motion within blocks is uniform and whilst a reduction in block size may assist in this consideration, this makes the computational overhead greater. Chan, Yu and Constantinides[7] describe a technique of variable-size block-based searching, where smaller blocks are allocated to more detailed areas of an image, especially where motion is evident. However, the process of segmenting the picture introduces new processing needs and the benefit of having less blocks becomes diluted.

The basic concept of variable-size block matching is, however, useful. Consider the image of figure 2(a). Here we see a head-and-shoulders scene, typical of audiovisual telecommunications, in which the background is largely stationary, with motion caused by the subject features contributing only a small part of the reconstructed image. However, as soon as the head moves from side to side (figure 2(b)), or the scene pans to include, say, another party, a significant amount of motion is taking place. This behaviour leads us to deduce that it will be of most use to concentrate only on areas where displacement is taking place and work at Nottingham has concentrated on the analysis of features in the image and the extent to which they change and are displaced over a given sequence.



(a)

(b)

Figure 2: Frames from the CCIR sequence *Miss America* showing (a) frame 58 (b) frame 76

2. FEATURE ANALYSIS

Motion estimation is a process requiring the clear definition of search parameters in a given algorithm and the primary failing of the block-matching motion compensation (BMMC) technique is that whilst it is applied throughout the image, the blocks are rather arbitrary with little contribution to the nature and content of features within the scene. Adding to this the drawback of BMMC in not being able to represent anything other than simple displacements and its in-built processing redundancy, a method is to be found that can depict motion in primary features, since it is already known that outline parameters are of most interest when performing picture reconstruction[8] .

Feature analysis attempts to resolve many of these difficulties by moving away from a block-matching method towards an approach dealing with variable areas and shapes within the picture. Tests on humans and other primates[9] reveal that perception of motion is more due to the spatio-temporal energy of displacements in the edges of shapes, than to the observation of shifts by arbitrary blocks within a given frame. By specifying an image in terms of its feature content, we can easily map the outline of various patterns and, by assigning simple primitive codes to those patterns, displacement vectors depicting interframe motion can be developed.

2.1 Pre-processing

One major consideration at this point is that whilst feature information will prove very useful, the structure of an image may be of a very large number of small, interlocking shapes - rendering full classification as impractical. To investigate this, an algorithm was developed to uniformly quantise an image and then assess the distribution of interframe difference magnitudes, with respect to each frame in an image sequence. The results are most interesting and are plotted in figure 3, using a quantisation step interval of 8 grey levels. It is notable that a high concentration of "small" differences exists and their frequency of occurrence is found to diminish as the quantisation step interval increases. However, there is a set of "large" differences, whose frequencies of occurrence are directly related to *overall* motion in an image and it can be seen that for the sequence *Miss America*, head movement is most pronounced between frames 58 and 76, as shown in figure 2. From this information it can be deduced that major motion is indicated by the presence of such magnitude differences, which could be compensated by a displacement estimation algorithm, whereas the smaller magnitude differences are probably within the scope of coding by an interframe pixel approach, such as the DPCM/DCT algorithm.

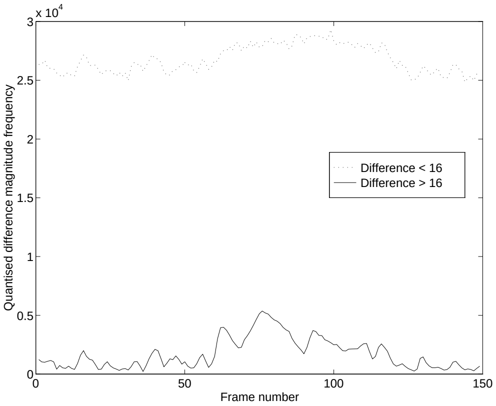


Figure 3: Distribution of difference magnitudes for the *Miss America* sequence

Given these results, quantisation can be deliberately introduced to images being processed for motion compensation, provided it is known that the DPCM/DCT algorithm will safeguard against the long-term propagation of errors.

A further approach to the overall reduction of image feature content is the introduction of regular blocking, which lowers the resolution and thus removes completely single-pixel features that would prove impossible to relocate on subsequent frames. Whilst this inevitably affects picture quality, the use of a scheme taking the mode value of pixels prior to blocking ensures that no new coefficients are introduced.

Both uniform quantisation and mode blocking have been successfully applied as pre-processing for the feature analysis algorithm. The result (figure 4) is a low resolution image, containing a reduced set of blocked features having only horizontal and vertical sides.

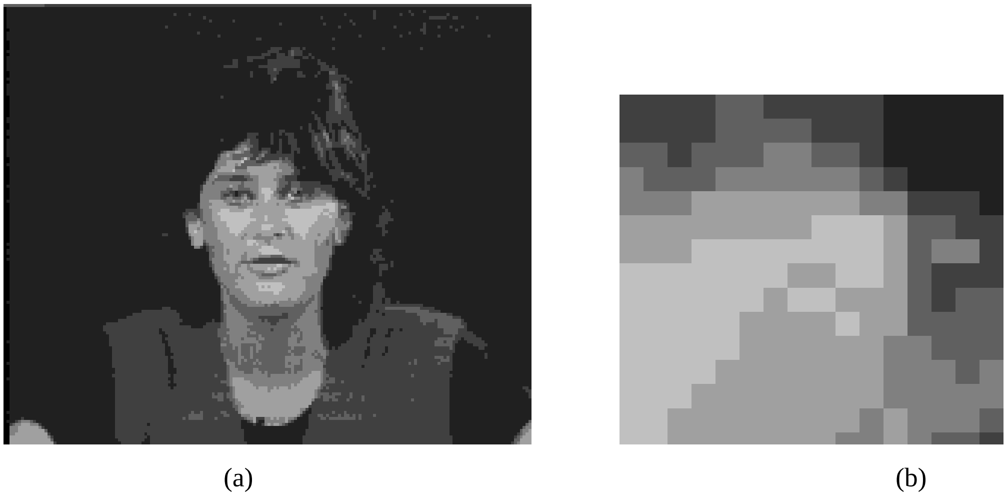


Figure 4: Resulting low-resolution image with (a) quantisation and mode-blocking applied, showing (b) the nature of blocked features

2.2 Feature Classification

The derivation of feature descriptors for each frame forms the reference set used as a look-up directory for the generation of displacement vectors. A clustering algorithm has been developed that groups neighbouring mode-value blocks of equal magnitude, taking as an origin the point first encountered in a scan from the top left-hand corner of the image to the bottom right-hand corner. With a cluster defined, a run-length of signed magnitudes in the real and imaginary axes describes the outline of the feature. By recording the origin, runlength descriptor and mode value, we have an efficient mechanism for the comparison of shapes between frames, using look-up tables to seek affine values in each of the three feature parameters (figure 5). The fundamental benefit of this approach is that whilst fairly intensive for smaller areas of motion detail, the effect of pre-processing ensures that large features of constant mode value, such as the stationary background of figure 2, use no greater degree of description. As will be described later, static features can soon be noted, allowing the algorithm to concentrate on local feature classification for areas of regular motion.

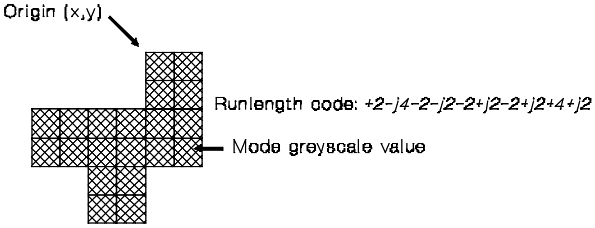


Figure 5: Generation of sourcebook entry for a cluster of equal mode-block values.

One further aspect of feature classification is the scope for the use of a shorthand notation for frequently-occurring primary shapes. Here, in place of a full runlength descriptor, a simple code can be used to describe the shape, together with a coefficient specifying an enlargement from unit-distance values. Figure 6 demonstrates how a t-shape primitive, expressed in unit-distance values, can simply be multiplied to any larger shape of the same outline. The nature of the adaptive algorithm being developed ensures that whilst some primitives, particularly simple rectangles, may be held in permanent look-up directories, others found to be popular in a given sequence may subsequently be added, with conversion made between the runlength descriptor and the primitive code. This technique has proved useful, since it reduces the extent to which runlengths need to be re-constructed in order to deduce interframe relationships.

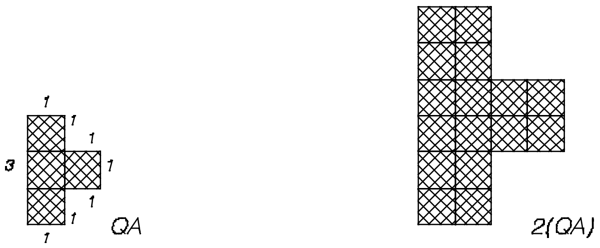


Figure 6: Description of shapes as multiples of primitive unit-length features.

2.3 Displacement analysis

With a method of feature classification defined, emphasis was next given to the application of sourcebook searching, given the different types of displacement that may arise. Motion may broadly be divided into two categories - *uniform*, where all points in a feature are displaced by a vector of constant magnitude and direction, such as linear translation and changes in perspective and *complex*, where vectors are inconsistent, such as for rotation and perspective skewing. Consideration of these factors further invalidates the efficiency of BMMC techniques, where only linear translations can effectively be described.

The feature classification technique, in conjunction with the quantisation and mode-blocking techniques employed for pre-processing, inherently allows for complex motion to be described, simply because it can be dealt with in discrete steps - a side effect of the induced low resolution. The feature analysis algorithm can compensate for rotation in multiples of 90^{o}, where the runlength descriptor is similar to the original shape, but where the origin has shifted to another corner of the feature. Figure 7 shows how the runlength descriptor, whilst fixed in magnitudes, will alternate between real and imaginary axes during a 90^{o} shift, as the origin moves along the sequence of edge values. Whilst this method of rotation measurement appears crude, the nature of the low resolution source image is such that the mode blocking and quantisation will remove interpolation between 0^{o} and 90^{o} rotation.

Perspective changes are ideally compensated by the feature analysis algorithm, as it is known that the minimum size of any given feature is that designated by the existence of unit-distance values. Hence, in the case of a primitive, the multiplying factor will simply be reduced or, for a runlength, the set of values will lower towards unit distance at the same rate. Perspective changes are only assumed where a given shape at its original value is no longer to be found.

Complex motion, particularly rotation, cannot be described by a single vector and the algorithm hence employs a vector pair to map both the destination of the first origin and the source of the new origin (figure 7). Where a single vector is provided, the decoder assumes a uniform displacement and the maintenance of the origin with respect to the shape.

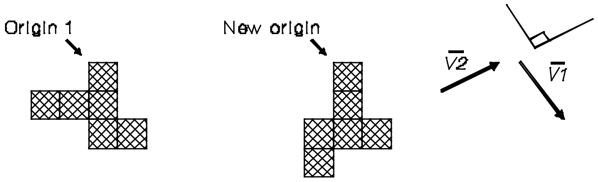


Figure 7: The generation of vector pairs describing rotational displacements.

The result of each stage of processing is an algorithm that can perform motion compensation, using the minimal computational overhead required for a classification method. It also accounts for changes in perspective and for rotation, by the introduction of a further vector.

3. IMPLEMENTATION

The feature analysis algorithm was implemented in the laboratory using 150 frames of the CCIR's Common Intermediate Format[¶] sequence *Miss America*, in a single band of the colour image. Quantisation was applied throughout the sequence and mode-blocking performed, choosing 2 × 2 original pixels as the mode-block size. The result of this was an image reduced in resolution to 25%. The quantisation step interval was set uniformly to 32 grey levels, given the constraints of the signal-to-noise ratio (figure 9) throughout the sequence and the desire to retain the large-difference characteristics discussed in section 2.1.

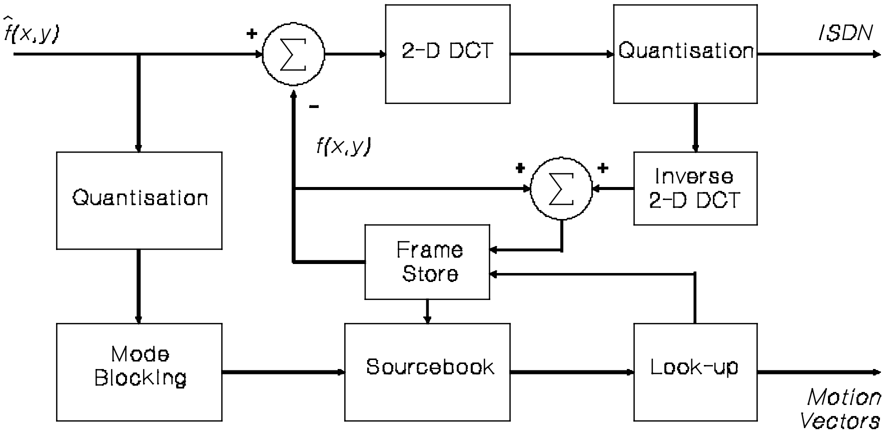


Figure 8: Incorporating variable shape motion compensation with part of the CCITT H.261 algorithm

3.1 Sourcebook generation and analysis

For each frame a sourcebook is generated, using a clustering algorithm for adjacent mode-block values, resolved into either runlength descriptors or primitive codes. The distribution of feature runlength is directly governed by the level of quantisation introduced to an image and figure 10 shows the frequency distribution of features, based on the

[¶] The Common Intermediate Format (CIF) is a video standard of 360 × 288 pixels, 30 frames per second, used by the CCITT to overcome standards conversion between NTSC and PAL.

quantity of constituent blocks, for a single frame of the sequence. Compared with a quantisation step interval of five grey levels, it can be seen that there is a greater number of large clusters, hence reducing the overall size of the sourcebook produced. The primary benefit of a small look-up sourcebook is an overall reduction in the time required for local searching.

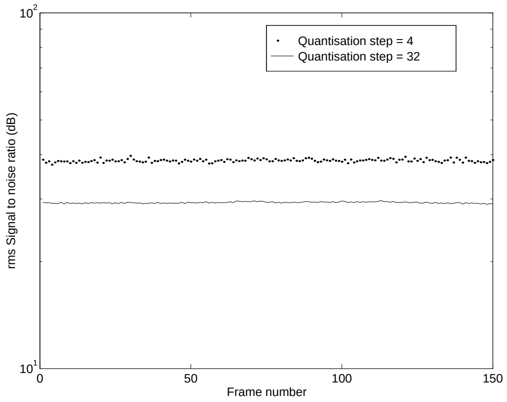


Figure 9: Signal to noise ratio for the re-constructed sequence

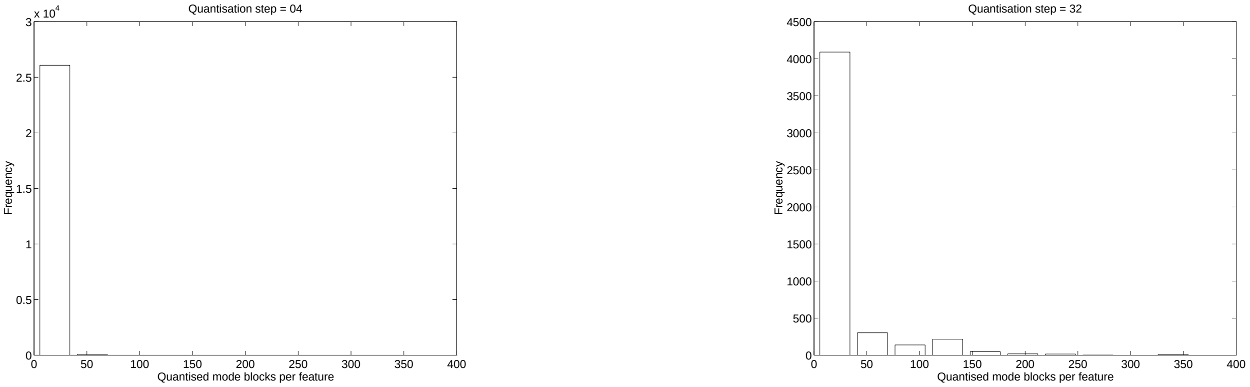
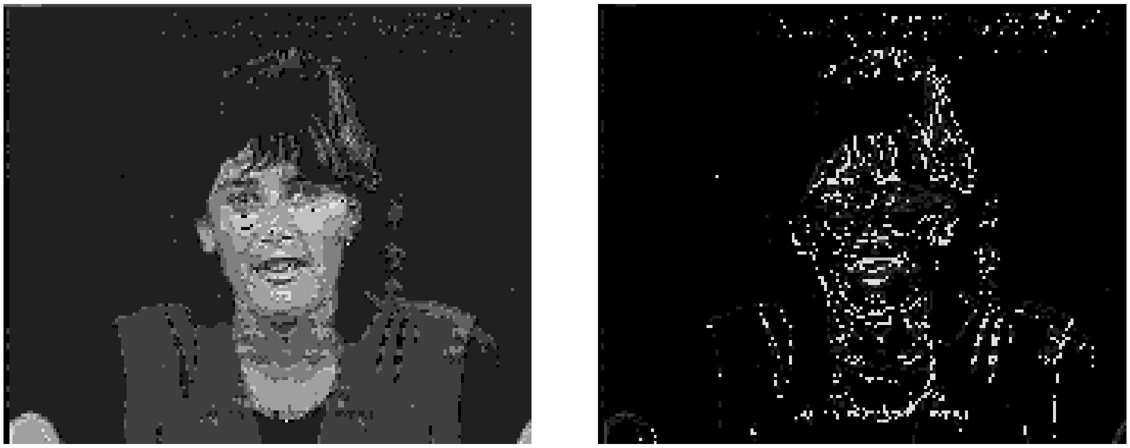


Figure 10: Feature size frequency distribution by clustered blocks

3.2 Vector decoding

Vectors produced by the local search process are received by the decoder as either singles or vector pairs. Uniform displacements are simple to re-map, as the origin and feature runlength may simply be translated on the reconstructed image. Complex displacement is, however, more involved as it has been found that the process of rotation is more likely to cause distortion in the runlength sequence than is the case for uniform motion. Whilst this exists as a limitation of the algorithm, in practice no decoding defects have been observed in the test sequences, suggesting either that rotation in general is not a normal feature of audiovisual scenes, or that overall rotation of large bodies is practically compensated as a series of uniform displacements for each of their smaller constituent features. Indeed, it was noted that vector pairs account for an average of only 2% of all vectors produced for the *Miss America* sequence.

Figure 11(a) shows a single frame from the sequence, with quantisation and mode-blocking applied prior to vector generation, together with a reconstructed frame using only motion compensation. Close inspection reveals a number of spatial defects, particularly where the algorithm has interpolated the motion of features, leaving unassigned areas of the reconstructed picture, figure 11(b). However, as a technique of motion compensation to be used in conjunction with a pixel differencing algorithm, analysis of the signal to noise ratio throughout the sequence clearly demonstrates that for the hybrid coding algorithm, error propagation is not an issue. For the purposes of experimentation, the decoded image was merged with DPCM values from the original image sequence, providing the required interpolation and keeping error levels to a minimum. A plot of the rms signal to noise ratio for the reconstructed sequence, compared with original values, is shown in figure 9. It can be seen that whilst fluctuations do occur between frames, the overall trend of errors is unchanging, for each level of quantisation.



(a) (b)

Figure 11: Re-constructed frame with (a) motion compensation showing (b) error distribution

In the simulation of these results, the full H.261 algorithm has not been employed, as only DPCM values are required for reconstruction. However, in practice a codec would subsequently perform a discrete cosine transform on blocks of pixels and quantisation of the transform coefficients (figure 9), which would add a further source of error at the decoder.

3.3 Computational efficiency

It has already been shown that the BMMC algorithm is inefficient at real-time rates and as the variable-shape approach described in this paper is not exhaustive, it is probably of little use to make direct comparisons of either's efficiency. However, both methods are variable-resolution techniques and where the block size is greater for BMMC, a good representation of motion in a scene can be made, although the spatial resolution of the vectors is low. By applying quantisation to the variable-shape approach, we can reduce the frequency distribution of features up to a level at which resolution is low enough to make reconstruction difficult.

Results obtained elsewhere[10] have shown that the uncompressed motion vectors produced by block-matching motion compensation for the Miss America sequence, vary from 0.172 bits/vector for no compensation, to 0.092 bits/vector for 8 × 8 pixel blocks, where each of these values is a ratio of the total data rate to the quantity of vectors produced. The variable shape algorithm produces a bit rate in the order of 0.05 bits/vector, however it should be noted that this is primarily due to the plain background minimising the quantity of motion vectors. Whilst encouraging for this particular sequence, the addition of more detail will add to the quantity of motion vectors produced, irrespective of which quantisation level is employed.

One enhancement to the variable shape algorithm is the elimination of null vectors for static features. Here, instead of transmitting empty vector sets to confirm that features are stationary, frequently occurring null vectors can be phased out over the course of a sequence, hence only sending data relating to field activity. This will further reduce the quantity of displacement data produced and cause a subsequent reduction in motion vectors.

4. APPLICATION TO SINGLE IMAGE INTRAFIELD COMPRESSION

For the purposes of real-time coding, the feature analysis algorithm has employed quantisation and mode-value blocking to reduce picture resolution and decrease the extent to which vectors are generated. However, it is clear that the algorithm could be further refined to make feature description and classification feasible as an efficient mechanism for single image compression. At present, most single image compression schemes, such as those developed by the Joint Photographic Experts Group (JPEG), employ intrafield line runlength-redundancy removal - a method analogous to DPCM, except here the relationships are spatial rather than temporal.

The runlengths produced by the feature analysis clustering algorithm can equally be applied in this way and although the absence of quantisation and mode-blocking will significantly increase the quantity of features produced, this is not a drawback for a technique that has no real-time constraints.

5. CONCLUSIONS

This paper has introduced a new method of interframe motion compensation, using displacement vectors generated by the analysis of variable shapes between frames. The method can compensate for both uniform and complex motion in varying degrees of efficiency. Whilst errors are introduced by way of quantisation and mode-value blocking, it has been shown that the net propagation of errors in a sequence is not evident, where the algorithm is used as part of a hybrid video codec also employing interframe pixel differencing and transform coding.

Refinements to the technique will develop an approach that can efficiently interpolate for complex motion such as rotation and skew. However, it will be essential to ensure that the introduction of additional processing does not excessively add to the computational overhead of the codec.

It has also been proposed that the feature analysis technique can be employed as a single-image compression algorithm, where no pre-processing is employed. However, it is as a motion compensation system that feature analysis has been successfully employed for video coding.

Acknowledgements

This research is funded by the United Kingdom Engineering and Physical Sciences Research Council. We are grateful to Mr David Ingham of the Department of Computer Science, University of Newcastle upon Tyne for his advice with software issues and to Mr Chris Greenhalgh of the Communications Research Group, University of Nottingham for help in preparing the videotape which accompanies this paper.

Bibliography

1. CCITT Recommendation H.261 ‘Video codec for audiovisual services at p × 64 kbits/s’, July 1990

2. Carr M. D., ‘Video codec hardware to realise a new world standard’, *Br Telecom Technol J* vol 8 no 3, July 1990 pp 28-35

3. Jain J. R. and Jain A. K. ‘Displacement measurement and its application to interframe coding’, *IEEE Trans* **COM-29** (12), 1981 pp 1799-1808

4. CCITT ‘Description of reference model 6 (RM6)’, SGXV, Specialists Group on Coding for Visual Telephony, Doc 396, 1988

5. Parke I. and Morrison D. G., ‘A hardware motion compensator for a videoconferencing codec’, *Proc IEE Colloquium on Motion Compensated Image Processing*, 1987, pp 1/1-1/5

6. De Vos L. et al ‘VLSI architectures for the full-search block matching algorithm’, *Proc ICASSP ’89*, **M5.22**, May 1989, pp 1687-1690

7. Chan M. H., Yu Y. B. and Constantinides A.G., ‘Variable size block matching motion compensation with applications to video coding’, *Proc IEE* vol 137 pt 1 no 4, August 1990 pp 205-212

8. Seferidis V. and Ghanbari M., ‘General approach to block matching motion estimation’, *Optical Engineering* vol 32 no 7, July 1993, pp 1464-1474

9. Sekuler R. and Blake R., *Perception* 3rd ed. McGraw-Hill, 1994

10. Cordell P. J. and Clarke R. J., ‘Low bit rate image sequence coding using spatial decomposition’, *Proc IEE*-I vol 139 no 6, December 1992, pp 575-581