CLASSIFIED SUBREGION MOTION ESTIMATED INTERFRAME CODING

N W Garnham and M K Ibrahim[¶]

Introduction

This paper introduces the framework of a novel algorithm for the detection, classification and identification of image subregions. It is shown that an efficient technique can be implemented for the representation of interframe motion in a video sequence. Whilst the method is not exhaustive, it is believed that it can be realised as an original approach to motion vector coding in a simple, low-resolution video codec.

Motion estimation and compensation

Motion estimation has, for several years, been associated with displacement compensation in predictive hybrid video codecs. In conjunction with the DPCM/DCT techniques employed as the basis of the ITU-T H.261[1] algorithm, motion compensation provides useful interpolation of difference data where significant interframe activity has occurred. Methods of motion compensated coding are mainly block-based, ranging from fixed-size full-search techniques, to more complex methods of feature extraction. Any novel displacement algorithm must be at once efficient and computationally of low complexity. Full-search techniques are useful, however they normally carry a high computational overhead and there is a considerable degree of process redundancy in sequences where motion is not distributed throughout the frame. Recent developments in variable block-size estimation show an apparent trend towards methods that could work independently, particularly where the images are not of high resolution.

Some work has considered the possibility of model-based interframe coding[2], where a form of feature recognition algorithm can be used to extract useful information about edges, changes in luminance and texture. In terms of image perception, this approach is more useful than simple block matching techniques, where vectors produced tend to be arbitrary in terms of the actual features they represent.

Classified subregion motion estimation builds on both these principles to produce a representation of motion that shows the displacement of regions, whilst employing classification to improve the efficiency of the prediction.

Interframe properties

Figure 1 shows two images from the standard test sequence *Miss America*. In some respects this is not typical of a videoconferencing scene, although it is a good demonstration of the type of foreground motion we might expect during the course of a conversation. Notice that over a period of eighteen frames, the head has moved to the right. Whilst changes in the shape of the lips and eyes will have occurred, many features have been displaced by a simple linear translation. It can be seen by closer inspection that groups of pixels have retained their shape and intensity, as well as the more obvious large features in the frame. Given this, and the fact that most the background is unchanged, we have developed a classification technique to compare regions of pixels in adjacent frames and determine motion.

[¶] Department of Electrical and Electronic Engineering
University of Nottingham

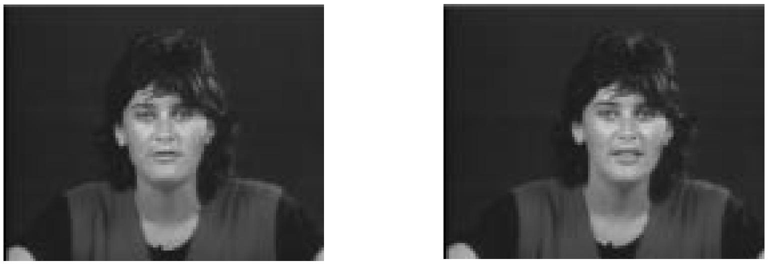


Figure 1: Frames 58 and 76 from the sequence Miss America.

Subregion identification

For the purposes of this work, a subregion is defined as being a group of pixels having the same intensity value, each of which has a horizontal or vertical relationship with its group neighbour. The algorithm implements this requirement using the four-neighbour principle:

$$\begin{array}{ccc} & q & \\ t & \boldsymbol{p} & r \\ & s & \end{array}$$

where q is vertically unit-distance adjacent to $\boldsymbol{p}$ and thus a member of the four-neighbour set $N_4(\boldsymbol{p})$. Subregions are identified by a scanning process, which starts from the top left-hand corner of the picture and works to the bottom-right. The first pixel encountered will be taken as the seed value of the first subregion, which will be grouped and the location of associated pixels recorded in a linked list. The process is repeated at the next pixel to be passed in the scan, which has not already been associated with a subregion. An illustration of this technique is shown in figure 2.

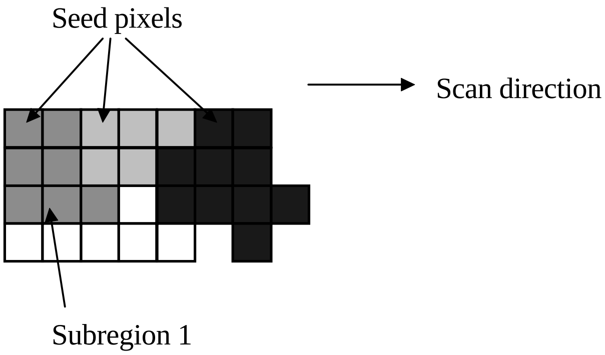
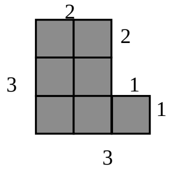


Figure 2: The identification of subregions

This process is repeated on each frame in the sequence, such that all frames contain known clusters of pixels, which are uniquely separate from other regions by pixel intensity. If we have up to n subregions in each frame, R, this relationship can be defined:

$$\bigcup_{i=1}^{n} R_i = R$$

where R_i is a connected subregion, $i = 1 ,2 ,..., n$
and $R_i \cap R_j = \varnothing$ for all i and j, $i \neq j$
where $\varnothing$ is the null set, demonstrating that adjacent subregions are disjoint.



the runlength will be +2 -j2 +1 -j1 -3 +3, with an area of seven blocks.

Whilst the use of perimeter runlengths provides a good representation, the data overhead in a video codec system would be quite high. To overcome this, a shorthand technique has been employed for frequently occurring primitive shapes. These range from simple squares and rectangles, to interlocking “T” and “L” shapes common throughout the image. Their classification is simple - all that is required is the pixel intensity, the shorthand code and the magnitude multiplier from unit values. Figure 5 shows how a *t*-shape is scaled from a unit primitive.

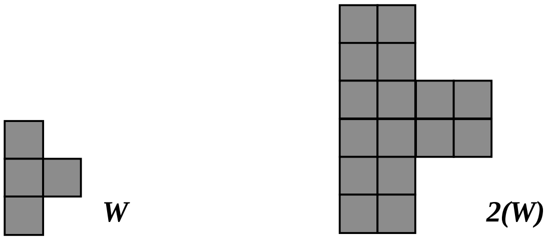


Figure 5: Shorthand notation for subregions scaled from unit primitives

Motion vectors

A list of classified subregions is stored in a look-up directory, which is supplied to the decoder. For the next frame, the process is repeated with the algorithm checking the directory for the presence of unchanged subregions at each seed pixel location. In most cases, there will have been no motion and no re-classification is required. Only where the match can no longer be made does the algorithm use a local search window to deduce the possible displacement of a previously identified subregion. When found, a simple motion vector, consisting of the origin and destination co-ordinates, can be generated, assuming the displacement is a linear translation. Where the match fails completely and no likely destination can be found, the decoder must be updated with new subregion descriptions in the area affected. Practically, it has been found that most displacements in the sequences *Miss America* are linear, with only a small proportion of shapes needing reclassification.

Figure 6 shows the location of non-zero motion vector origins, where subregion translations are able to account for interframe displacements. Considering the bitrate normally associated with interframe coding, it can be seen that only a small part of the picture is affected and in many respects the transmission overheads are similar to those connected with DPCM coding.

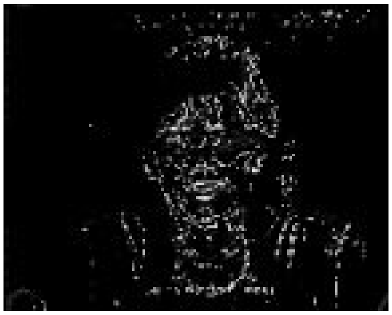


Figure 6: Location of non-zero displacement vectors

The *Miss America* sequence is useful for demonstrating the principle of subregion motion estimation, however it cannot be considered typical if the technique is to be applied to videoconferencing. To demonstrate a more practical scenario, consider the image of figure 7, from the *Salesman* sequence, to which we have applied the same pre-processing, with linear spatial quantisation set to eight grey levels.

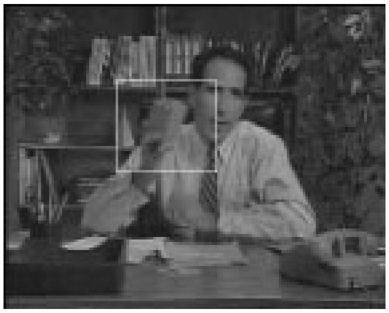


Figure 7: Pre-processed frame from the Salesman sequence.

In this case the background is more complex and the initial coding of subregion descriptors carries a more significant overhead than we have previously described. However, the background is stationary and, once coded, all its constituent subregions can be represented by null vectors. The viewer will take most interest in the moving object being rotated by the salesman, which coincides with the area of most estimation activity. The algorithm has been developed to be self-learning, detecting trends in the displacement of regions, so that whilst most of the image will comprise unchanging shapes or translational motion, it can be determined where re-classification of features is likely to occur.

Error detection and correction

Whilst reclassification is available to the codec, it may be the case that a subregion has altered in only a small part during interframe motion. The result of this can be detected by small voids in the reconstructed sequence where insufficient information exists to remap the change in topography. To overcome this, we have used a simple adaptive filter which locates these errors and performs mode-value filtering using a small template of typically 3 × 3 pixels, as shown in figure 8.

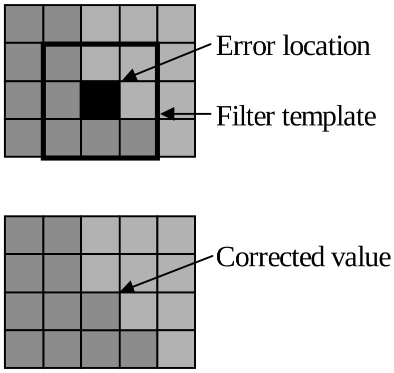


Figure 8: The correction of interframe displacement errors.

Whilst this is best appreciated in a video sequence, where temporal quality is more obvious to the viewer, the image of figure 9 shows how the application of this technique can provide an image without the distraction of such errors.

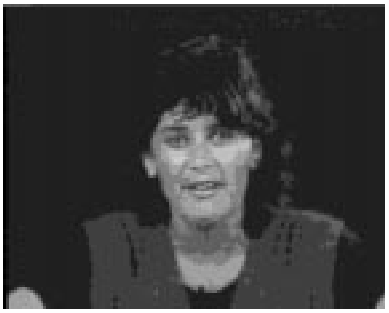


Figure 9: Reconstructed image following error filtering

Conclusions

The application of classification to image subregions represents a departure from most contemporary thought which is directed to the use of block-based coding in hybrid codecs. The classification process requires no degree of intraframe recursion and the generation of look-up directories is a simple process, particularly where known primary shapes can be represented by shorthand notation.

Transmission coding of vector data can employ variable length codes, developed using the conventional Huffman techniques. Once classified, stationary objects can be represented at the decoder using null vectors, coded using the most efficient datagram. Only where reclassification occurs, does a greater overhead on the transmission system apply.

Future work will continue to optimise the method of feature representation. The application of neural networks will inevitably improve the scope of these methods for real-time coding, where larger subregions can be broken down into smaller areas conforming with primary shape criteria. Again, this is a compromise between the use of more vectors to describe interframe activity and the longer descriptors to represent the changing topography of subregions.

References

1 ITU-T Recommendation[†] H.261, Video Codec for audiovisual services at p × 64 kbit/s, rev 1993
[†Previously “CCITT Recommendation”]

2 Welsh, W.J., ‘Model-based coding of videophone images’, *IEE Electronics & Communication Engineering Journal*, Feb 1991, pp 29 - 36

3 Treismann, A, ‘Features and objects in visual processing’, *Scientific American* no 255, 1986, pp 114-125