Extending Lossless Image Compression

Andrew J. Penrose
Andrew.Penrose@cl.cam.ac.uk

Neil A. Dodgson
Neil.Dodgson@cl.cam.ac.uk

Computer Laboratory
University of Cambridge

Abstract

We develop a lossless compression scheme for colour video that takes advantage of the spatial, spectral and temporal redundancy inherent in such data. We show that an adaptive scheme is vital to ensure full exploitation of these redundancies for compression purposes. The results of the proposed scheme are found to be favourable when compared to lossless image compression standards and this performance is achieved while still permitting a computationally simple decoder.

1 Introduction

Compression of colour images and video is ubiquitous in today's multimedia world. Such compression schemes are mostly lossy, with information from the original source sacri£ced to obtain extra compression. While the extra compression gained by such methods usually outweighs the image degradation that results from lossy compression, some applications still bene£t from, or require, lossless compression. Such applications include scienti£c and medical image storage and high quality video production techniques.

Lossless image compression has traditionally been based strongly on the compression of greyscale images via predictive coding[1]. Colour images and video have generally been treated as a set of unrelated greyscale planes and a sequence of unrelated frames respectively. However, some techniques have been reported in the literature that use the extra redundancy in colour images and video to obtain extra compression [2, 3, 4, 5]. These extended schemes gain extra compression at the expense of increased computational complexity. While this may be acceptable in some circumstances, it limits the possible applications. Consider an archive of colour video; it is likely that the video data will be compressed once and decompressed many times. Such a system would clearly bene£t from the ability to decompress (playback) the archived video in a time sensitive fashion and this in turn would be made most convenient by a scheme with a computationally simple decoder.

In the rest of this paper a lossless colour video compression scheme is developed, based on predictive coding, which takes advantage of the extra correlations present and yet permits a computationally simple decoder. In Section 2 extensions to the standard predictive model are discussed, while in Section 3 adaptation and symbol coding are introduced. The results of the proposed scheme are detailed in Section 4 and conclusions and further work are listed in Section 5.

Eurographics UK '99, 13-15 Apr 1999, Fitzwilliam College, Cambridge, ISBN 0-9521097-8-6

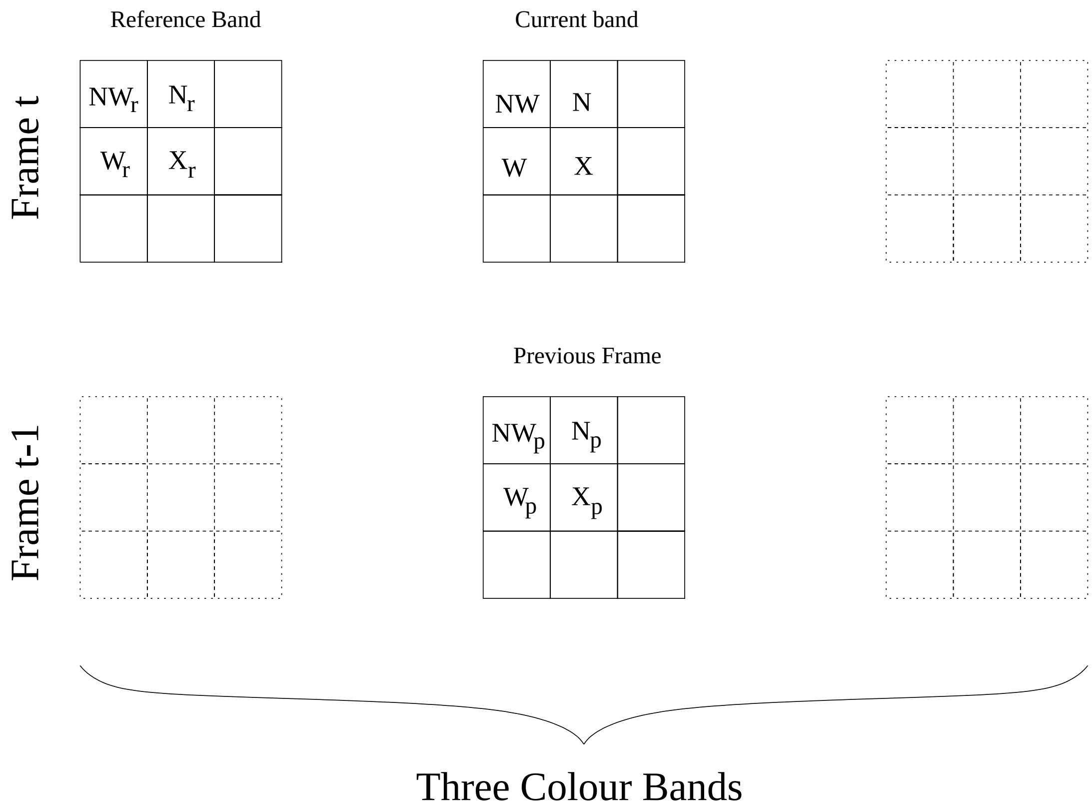

Figure 1: Labelling of spatial, spectral and temporal locations around the current pixel **X**.

2 Greyscale and Beyond

Predictive coding of greyscale images typically considers each pixel in raster scan order. A prediction of the current pixel value X is made based on neighbouring pixels such as W, N and NW (see Figure 1). The error in such predictions tends to form a two-sided Laplacian distribution, sharply peaked at zero, making it amenable to ef£cient storage via symbol coding techniques (e.g. Huffman coding). By using only previously scanned pixels to make a prediction, the decoder is able to make the same prediction and can then use the stored prediction error to reconstruct the original pixel.

Many spatial (intraband) predictors have been suggested in the literature and one that has proved popular in recent times is the Median Adaptive Predictor (MAP)[6] which was used to good effect in LOCO-I[7]. The MAP calculates the predicted value for the current pixel, $\hat{X}$, by:

$$\hat{X} = \begin{cases} \min(W,N) & NW \geq \max(W,N) \\ \max(W,N) & NW \leq \min(W,N) \\ W+N-NW & \text{otherwise} \end{cases} \quad (1)$$

The MAP can be seen as selecting between three linear predictors based on a simple function of surrounding values. As discussed in [7], the MAP gives good prediction even in the presence of edge features.

2.1 Using Colour

In order to gain improved compression by using the extra information present in the correlation between different colour bands, a spectral (interband) predictor is needed. The result of many experiments by the authors suggests that the simple, yet effective interband predictor presented in [4] is the best currently known. This predictor estimates $\hat{X}$ using neighbouring intensity values from the current colour band (W and N) and from a reference band (X_r, W_r and N_r):

$$\hat{X} = \frac{W + (X_r - W_r) + N + (X_r - N_r)}{2} \qquad (2)$$

This can be seen as using the horizontal and vertical intensity gradients in the reference band, to model the same gradients in the current band. It should be noted that we require the decoder to know the values of pixels in the reference band, therefore not all image data can be predicted spectrally.

2.2 Moving Through Time

The simplest of all temporal (interframe) prediction schemes is to use the intensity value from the same location in the previous frame, that is $\hat{X} = X_p$. This predictor is based on the assumption of a static scene and works best when there is no camera or object motion.

To combat the effects of object and camera motion, most video coding schemes (which are lossy) use *motion compensation*. Motion compensation requires the encoder to perform motion estimation, which gives an estimate of the motion between frames. These motion estimates (motion vectors) are sent as overhead to the decoder and can be used to produce a temporal prediction that is superior to $\hat{X} = X_p$.

However, motion compensation is computationally expensive. Furthermore, work by one of the authors[8] has shown that by carefully selecting between simple spatial and temporal predictors, the performance of motion compensation can be nearly matched for greyscale video. So, $\hat{X} = X_p$ can reasonably be used as a temporal predictor in conjunction with other spatial and spectral predictors. Mechanisms for selecting between these predictors are discussed in the following sections.

It is important to note that temporal prediction, requires a reference frame. In theory, temporal prediction can be used for all but the £rst frame in a video sequence. However, in practice this would mean that to decode the n^{th} frame in a video, all previous frames would need to be decoded. Hence, in this work, as in others, independent frames (I-frames) that do not use temporal prediction are positioned regularly through the video sequence. This makes fast forward and rewind features feasible.

2.3 Comparing Predictor Performance

One way to gauge the performance of the predictors mentioned above is to measure the entropy of the prediction errors they produce. The entropy, in bits per symbol, is calculated by:

$$\sum_{i \in \text{possible errors}} p(i) \log_2 \left(\frac{1}{p(i)} \right) \qquad (3)$$

Entropies calculated in this way represent the theoretically optimal compression, given the assumption that prediction errors are independent and identically distributed (i.i.d.).

Video	No prediction	Spatial	Spectral_R	Spectral_G	Temporal	Adaptive
Claire	18.36	8.61	8.26	8.28	8.53	7.75
Football	21.35	15.15	14.05	14.00	18.17	13.90
Granny	21.55	10.16	9.59	9.27	8.84	6.92
Missa	18.35	12.21	11.31	11.24	13.83	11.18
Mobile	21.06	14.93	13.12	13.06	16.17	13.05
Susie	20.41	12.00	10.33	10.22	13.85	10.20

Table 1: Entropy of prediction errors in bits per pixel.

The entropies of the prediction errors, produced by applying various predictors to each of six test sequences, are given in Table 1. The columns show the results for no prediction (for reference), spatial prediction as in Equation 1, spectral prediction as in Equation 2 using red and green as reference bands (Spectral_R and Spectral_G respectively) and temporal prediction using $\hat{X} = X_p$. The £nal column gives results from an adaptive scheme which is described in the following section.

For spectral prediction, the reference colour band was predicted spatially. Also, for temporal prediction every twelfth frame (starting with the £rst) was made independent by using spatial prediction.

Most of the six video sequences are widely used for test purposes and are available online (ftp://ipl.rpi.edu/pub/), the exception being the *Granny* sequence. *Granny*, unlike any of the other sequences, is computer generated and consists of both shaded and textured surfaces. All the sequences are 24 bit colour (8 bits each for red, green and blue). The results presented are based on the £rst 100 frames from each sequence, with the exception of *Football* and *Mobile* for which only 97 and 40 frames respectively were available.

The results in Table 1 show that spectral prediction is consistently better than spatial prediction. With the exception of *Claire* and *Granny*, temporal prediction gives the lowest performance for any predictor. In the case of *Football*, for which temporal prediction is particularly poor, this may be explained by a combination of camera movement, large object movement and artefacts caused by poor de-interlacing of the original footage. The best prediction performance was consistently achieved by the adaptive prediction scheme, which we now discuss.

3 Adaptation

It is in the nature of images for their properties to vary over their extent. Thus a predictor which works well for some parts of the image data may not work so well on other areas. For a particular video sequence, temporal prediction may work well for a static region in the scene, whereas spectral prediction may be more effective in an area of motion. If both spectral and temporal correlations are poor, spatial prediction could be used instead.

This concept of predictor selection was tested alongside the individual use of the other prediction methods. Each frame in the video sequence was divided into blocks (5 by 5 pixels) and the *Mean Square Error* (MSE) for each predictor was calculated in each block. The predictor with the lowest MSE in a given block was chosen for that block. The results from this scheme, shown in the £nal column of Table 1, include the overhead of recording the predictor selection decision. Even with the overhead, this adaptive scheme gives superior performance

Symbol	$k=0$	$k=1$	$k=2$
0	*0*	0*0*	00*0*
1	*10*	1*0*	01*0*
2	*110*	0*10*	10*0*
3	*1110*	1*10*	11*0*
4	*11110*	0*110*	00*10*
5	*111110*	1*110*	01*10*
6	*1111110*	0*1110*	10*10*
7	*11111110*	1*1110*	11*10*

Table 2: Examples of Golomb-Rice codes for various k. The bits in italics show the unary section of the code.

to the use of any individual predictor.

However, the MSE of prediction errors does not necessarily have a one to one relationship with the number of bits used to represent those prediction errors. Predictor selection would be most effective if the decision made minimised the actual number of bits used in coding. Thus, predictor selection is best considered alongside symbol coding.

3.1 Symbol Coding

After prediction, a symbol coder is needed to ef£ciently store the prediction errors. Just as predictor ef£ciency varies over an image, so the ideal coding parameters change as well. Indeed, adaptive symbol coders are universal in modern image compression. Taking what has been said so far, we require a encoding scheme with the following features:

- It adapts easily.
- The number of bits required to encode a symbol is readily computable.
- It decodes quickly.

Such a scheme is Golomb-Rice coding, as used in [7, 9]. These codes, which are a subset of the Huffman codes and optimal for a Laplacian distribution, require a single parameter k. To encode a symbol, the k least signi£cant bits are used with the remainder of the symbol sent as a unary sequence. A few example codes are given in Table 2.

Thus, in order to adapt the code to varying statistics, only a suitable value of k needs to be selected. The number of bits required to encode a symbol x with a Golomb-Rice code of parameter k is:

$$k + \left\lfloor \frac{x}{2^k} \right\rfloor + 1 \tag{4}$$

This number can be quickly calculated. Thus, the co-jointly optimal parameters for predictor selection and symbol coding can be quickly calculated by counting how many bits would be output for each possibility and choosing the minimum.

Most modern lossless image compression schemes use *context* based adaptation[7, 10, 11]. Such schemes calculate a context for the current pixel based on neighbouring pixels. Statistics gathered in previous instances of the current context are used for adaptation. Because

Video	LJPEG	JPEG-LS	Proposed
Claire	9.40	7.40	6.95
Football	15.55	10.50	13.33
Granny	11.18	8.96	6.54
Missa	12.88	11.26	10.98
Mobile	15.55	14.08	12.91
Susie	12.45	11.26	10.26

Table 3: Actual compression performance in bits per pixel.

this form of adaptation is based only on what has already been seen it is often called *backward* adaptation.

However backward, context based adaptation requires both the encoder and decoder to calculate contexts and for each context maintain statistics of past prediction errors. An alternative scheme is to use forward, block based adaptation, as used for the adaptive predictor selection mentioned above. By letting the encoder look forward over data that is yet to be coded and then sending the results of the adaptation to the decoder as overhead information, the computational load on the decoder is decreased. This comes at the potential loss of some compression performance. Such a tradeoff is deemed reasonable for a video archiving application, where the expectation is that data will be encoded once and decoded many times.

4 Results

A lossless compression scheme for colour video has been implemented, based on the above discussion. Four predictors are made available; spatial, spectral (using either red or green as reference bands) and temporal. Forward, block based (5 by 5 pixels) adaptation is used for predictor selection and for the choice of Golomb-Rice code parameter k. Three bits are used to transmit k, giving a range of 0 to 7. However, as larger values of k are rarely used, the value 7 is used as a special run event marker. Such a marker is used if all the prediction errors in the given block for the speci£ed predictor are zero.

The results of this scheme are compared with lossless image compression standards in Table 3. The columns labelled *LJPEG* and *JPEG-LS* show the results of using the old lossless JPEG[12] standard and the new lossless JPEG[9] standard respectively, on each individual frame of the test video sequences.

We see that the proposed scheme offers the best compression for each sequence, except *Football*. This could be attributed to the artefacts related to poor de-interlacing of the original source. The effects of this horizontal banding is apparently better modelled by the context based adaptation in JPEG-LS, than it is by the block based adaptation of the proposed scheme. Also, the proposed method gives compression results that mostly surpass the entropy calculations shown in Table 1. This is made possible by the adaptive symbol coding, which exploits the fact that prediction errors are not i.i.d. to gain extra compression.

It is worth mentioning another study which is similar to this work. In [5], Memon and Sayood describe a lossless scheme for colour video compression based on forward adaptive predictor selection and backward adaptive error modelling. The predictors and symbol coding methods both differ from those used in this work. Also, their predictor selection allows only spectral and temporal prediction; spatial prediction can only be used for the spectral reference band. The scheme developed in [5] is tested only on the *Football* and *Susie* sequences. The results in [5] show that less compression was achieved compared to the method proposed in this paper.

By studying the number of times each predictor was used, we £nd that only *Claire* and *Granny* make signi£cant use of the temporal predictor. This £nding is echoed in [5], which reports that spectral prediction is more important than temporal, at least for *Football* and *Susie*. A possible reason for the success of temporal prediction in the two cases mentioned, is a combination of no camera movement and low camera noise (zero in the case of the computer generated *Granny* sequence).

Although the proposed scheme has been designed to allow a computationally simple decoder, the current implementation only achieves 12 frames per second when decoding the *Claire* sequence on a fast workstation. However, it is hoped that with further developments full frame rate playback of losslessly compressed video will become feasible.

5 Conclusions and Future Work

This work has shown that the compression of colour video can be improved by considering spectral and temporal correlations as well as spatial redundancy. The ef£ciency of temporal prediction was found to be highly dependent on individual video sequences. In particular, high quality computer generated video can make signi£cant use of temporal prediction. Given the results from earlier work[8] that found temporal prediction to be more useful for greyscale video, we can conclude that the relatively poor performance of temporal prediction, for some sequences, is due to spectral prediction being more ef£cient than temporal.

Another £nding from this work is that the extra compression available from colour video can be achieved without necessitating a large increase in decoder complexity. Indeed the presented scheme has a decoder that is less complex than many lossless image compression decoders, due mainly to the use of forward rather than backward adaptation.

Although this study considered a relatively large set of test video sequences compared to other such studies, more test sequences are needed to determine the extent of sequences for which temporal prediction is more ef£cient than spectral prediction. In particular, no medical or scienti£c time sequences were studied.

This study has assumed that it is acceptable to sacri£ce some compression to allow for fast decoding. In other applications, maximum compression may be the driving motive. Hence, future work will look at using alternative adaptation mechanisms and more aggressive error modelling to achieve higher compression.

References

[1] N. Memon and K. Sayood. Lossless image compression: A comparative study. *Proc. SPIE Still-Image Compression*, 2418:8–20, March 1995.

[2] N. Memon and K. Sayood. Lossless compression of rgb color images. *Optical Engineering*, 34(6):1711–1717, June 1995.

[3] S. Assche, W. Philips, and I. Lemahieu. Lossless compression of pre-press images using a novel color decorrelation technique. *Proc. SPIE Very High Resolution and Quality Imaging III*, 3308:85–92, January 1998.

[4] N. Memon, X. Wu, V. Sippy, and G. Miller. An interband coding extension of the new lossless jpeg standard. *Proc. SPIE Visual Communications and Image Processing*, 3024:47–58, January 1997.

[5] N. Memon and K. Sayood. Lossless compression of video sequences. *IEEE Trans. on Communications*, 44(10):1340–1345, October 1996.

[6] S. Martucci. Reversible compression of hdtv images using median adaptive prediction and arithmetic coding. *Proc. IEEE International Symposium on Circuits and Systems*, pages 1310–1313, 1990.

[7] M. Weinberger, G. Seroussi, and G. Sapiro. LOCO-I: A low complexity, context-based, lossless image compression algorithm. *Proc. Data Compression Conference*, pages 140–149, March 1996.

[8] A. Penrose. Lossless compression for video archiving. In *ACM Multimedia 98 (poster presentation)*, pages 76–77, Bristol, September 1998. Full presentation available via http://www.cl.cam.ac.uk/users/ajp1011/Abstracts/index.html.

[9] ITU-T Rec. T.84 - Information Technology. Digital compression and coding of continuous-tone still images: Extensions, July 1996.

[10] X. Wu and N. Memon. Context-based, adaptive, lossless image codec. *IEEE Trans. on Communications*, 45(4), April 1997.

[11] M. Weinberger, J. Rissanen, and R. Arps. Applications of universal context modeling to lossless compression of gray-scale images. *IEEE Trans. on Image Processing*, 5(4):575–586, April 1996.

[12] G. Wallace. The jpeg still picture compression standard. *Comms. of the ACM*, 34(4):30–44, April 1991.