A Rate Estimation Framework for Matching Pursuits Video Coding

Haoxiang Zhang, Adrian N Evans and Donald M Monro
Department of Electronic and Electrical Engineering, University of Bath, Bath, UK
Email: {eephz, A.N.Evans, D.M.Monro}@bath.ac.uk

***Abstract*— The Bath University Matching Pursuit (BUMP) project aims at developing new matching pursuit (MP) algorithms for still image and video compression. Compared to traditional MP coding, BUMP has many new features that have been designed with the aim of producing an efficient MP representation with reduced complexity. In this paper, we study in depth the characteristics the of bit rate change along the coding process of BUMP coder, and then propose a mathematical framework for describing the relationship between the number of coded atoms and the resultant bit rate. The framework has a very simple form and demonstrates very high modeling accuracy, which allows a precise estimation of the number of bits to be used throughout the coding process. We then present an adaptive algorithm for predicting the number of bits for use in our hybrid wavelet/MP video coder. Experiments show that the proposed rate estimation scheme is able to provide an extremely accurate estimation for the BUMP based video coder.**

***Index Terms*—video coding, matching pursuits, bit rate estimation model, rate control**

I. INTRODUCTION

Matching pursuits (MP) [1] is a signal processing technique that has stimulated much recent research. In particular, it has proved to be a powerful alternative tool to transform coding for visual signal compression [2]. The iterative nature of the MP process also has much potential for the development of simple and accurate rate control schemes for video compression.

A. Matching pursuit (MP) and MP based video coding

Matching pursuit was introduced as a signal processing algorithm by Mallat and Zhang [1]. It is a recursive process that iteratively decomposes a signal using basis functions selected from a predefined code book. The signal is then represented as a weighted combination of the chosen.

In hybrid video coding systems, motion estimation and compensation models are applied to video sequences to reduce their temporal redundancy, producing motion residual frames. Since a large part of the temporal correlation has been removed, motion residual frames generally exhibit a very sparse data structure, which is very well suited for encoding with MP algorithms. In [2], Neff and Zakhor introduced MP into video coding systems. In particular, they used MP to replace the DCT encoder for motion residual frames in an H.263 coding system. The new MP video coder was shown to be very effective, producing a significant increase in the PSNR performance over the H.263 system.

B. Bath University Mtaching Pursuit (BUMP) project

Neff and Zakhor's success in applying MP to video coding triggered much interest in MP based video coding schemes. Meanwhile, advances in conventional hybrid video coding area have been made, many of which are incorporated in the new video coding standard, JVT/H.264 [3]. H.264 is the state-of-the-art video coding standard and outperforms all its predecessors in the sense of coding efficiency [3]. Therefore, the incorporation of MP within an H.264 framework has much potential for improved MP-based video coding.

The aim of the Bath University Matching Pursuit (BUMP) is to develop visual signal compression schemes using MP-based algorithms. BUMP improves the coding performance by introducing some important features into MP, such as wavelet pre-transformation, the MERGE algorithm for encoding atoms [4], an efficient atom searching method [5] and new codebooks based on a sequential basis picking algorithm [5]. Also, it extends the MP algorithms to still image coding [4].

C. Proposed Rate-Atom analysis framework

Rate control is an important part of practical video coding. Rate distortion theories model the relationship between the coding bit rate and the signal distortion. Rate control and bit allocation schemes are then used to control the usage of bit budget and to optimize the system performance under coding constraints [7]. Many rate estimation models have been developed for conventional transform coding schemes and applied to their rate control [8] - [11].

Rate control is also a vital part for MP based video coders and has been the subject of some previous research [12]. The work mainly focuses on two parts: the first is how to adapt the quantization value in the video framework along the coding procedure. The second

---

Based on "A Hybrid Video Coder Based on H.264 with Matching Pursuits", by Haoxiang Zhang, Xiaopeng Wang, Wei Huo and D.M.Monro, which appeared in the Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 2006, Toulouse, France, May 2006. © 2006 IEEE.

aspect is the modeling of the relationship between the number of coded bits and atoms.

However, due to the different nature of coding methods exploited in BUMP project, the work of [12] cannot be directly applied to our coding scheme. In this paper, we study the characteristics of how the bit rate changes with the number of coded atoms. We define this as the Rate-Atom (R-A) analysis. A framework for modeling the bit rate of the MP video coder is proposed. This model is then used to design an adaptive rate estimation algorithm which is able to effectively track the R-A slope changes and ultimately provide accurate estimation.

D. Paper organization

The remainder of this paper is organized as follows. In section II, the matching pursuit algorithm is reviewed in the light of previous research on its application to video coding. The BUMP project is introduced in section III, focusing on the new features that distinguish it from other MP schemes. Section IV studies how the coding bit rate changes with the atom numbers within the BUMP codec. A new framework for modeling the relationship between the two values is given in section V and section VI shows how the framework can be used to design a rate estimation algorithm for the BUMP-based video coder. Experimental results and conclusions are presented in sections VII and VIII respectively.

II. THE MATCHING PURSUITS ALGORITHM AND ITS APPLICATION IN VIDEO CODING

A. Signal expansion with matching pursuits

The idea of matching pursuits is to decompose a signal into a linear expansion of waveforms, which belong to a redundant dictionary of functions $D=\{\varphi_\gamma\}$, with $\|\varphi_\gamma\|=1$ for all γ. MP is a recursive procedure in which, at each iteration, a function $\varphi_{\gamma n}\in D$ that best approximates part of the signal is chosen. So, after *m* iterations MP decomposes the original signal *f* into a sum of dictionary elements, so that

$$f=\sum_{n=0}^{m-1}\alpha_n\varphi_{\gamma n}+R^m f\,, \tag{1}$$

where a_n is the matching pursuit coefficient (or inner product) given by

$$a_n=\left\langle \varphi_{\gamma n},R^n f\right\rangle \tag{2}$$

and $R^m f$ is the *m*th order residual vector after approximating the signal in the direction of $\varphi_{\gamma n-1}$.

It is often impossible to obtain an exact reconstruction of the signal. Firstly, infinite number of iterations cannot be completed, and secondly, the matching pursuit coefficients are quantized to $\alpha_n' = Q(\alpha_n)$ prior to the calculation of the residual $R^{n+1}f$. Considering these two factors, the reconstruction of the signal is given by

$$f'=\sum_{n=0}^{p-1}\alpha_n{}'\varphi_{\gamma n} \tag{3}$$

B. Applying MP algorithms to video coding

The widely used hybrid video encoder employs motion compensation to predict the current frame from previously reconstructed frames. The predicted image is then subtracted from the current original image to produce the motion residual image. Since most of the correlation has been removed, a residual image often contains some sparse and clustered data structures. The mechanism of a MP coder is very efficient for encoding sources with sparse data distribution structure. Neff and Zakor first proposed the incorporation of MP coder into a hybrid video coding system [2]. This approach combines the advanced motion models of hybrid video systems with the high efficiency of MP coder and provides very good coding performance. The recently proposed BUMP video coder follows this approach and has proven to be very effective. Its structure is shown in Figure 1 [13].

III. THE BUMP PROJECT AND THE H.264 BASED MATCHING PURSUITS VIDEO CODER

A. The Bath University Matching Pursuit (BUMP) project

The Bath University Matching Pursuit project (BUMP) further develops the idea of MP coding. It aims to develop efficient matching pursuit algorithms for encoding video signals. For the purpose of improved compression efficiency, BUMP introduces many new features, including:

1) Designing new codebooks. MP algorithms try to decompose a signal by transmitting the approximation between the signal and the basis functions. Therefore, properly chosen bases are vital for providing a high coding efficiency. In BUMP, a special basis picking scheme is exploited to construct an effective codebook [5]. The codebook not only holds basis functions that have the ability of conveying the image information with high fidelity, but also has a very small size, which in turn requires fewer bits to encode the basis identity than a larger codebook.

2) Introducing a wavelet pre-transformation. The nature of MP makes it very effective at encoding data from sparse and clustered structures, such as residual frames. In our previous work [4] [13], it has been shown that by applying a wavelet transformation as pre-processing stage, the energy can be further concentrated so that the subsequent atom searching and coding processes can be more efficient.

3) Extending the MP coder into still image encoding [3]. From the same motivation as described above, we apply a wavelet transformation to still images so the energy is more focused and the data becomes sparse and locally clustered, and thus more suitable for MP encoding.

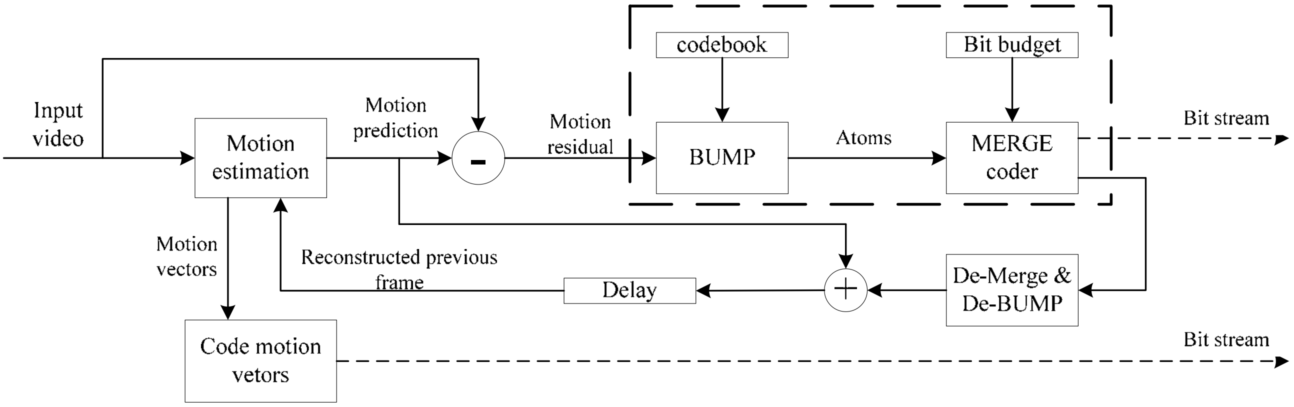


Figure 1. The hybrid video coder with BUMP and H.264.

4) Separable atom searching strategies. Various atom searching schemes are studied [5]. Some sub-optimal schemes allow a quicker searching procedure. The tradeoff between the performance and the complexity is studied and the best compromise is exploited.

5) Precision limited quantization (PLQ) for quantizing the atoms' amplitude. PLQ was first introduced for wavelet based image in order to design a visual lossless quantization method according to psycho-visual experiments [14]. Later it was found to be effective in helping to improve the compression performance of MP video coders [15]. PLQ provides an effective alternative quantization scheme for MP.

6) Multi-pass Embedded Residual Group Encoding (MERGE) coder for atom encoding. The MERGE coder classifies the atoms into groups according to their quantized amplitude and transmits them by encoding their positions. Together with PLQ, the BUMP coder is able to provide a perfectly embedded bit stream, and thus potentially allows a more accurate rate control for the video coder [13].

With all the above features, BUMP is able to provide MP visual compression algorithms of low complexity and high efficiency for various sources.

B. The hybrid video coder with H.264 and matching pursuits

In [13], the BUMP coding algorithm introduced above is applied to video coding, and a hybrid video coding system based on H.264 [16] with matching pursuit was proposed, with the structure shown in Figure 1.

The state-of-the-art H.264 standard represents the latest advances in video coding and outperforms most video coding schemes presented in the literature. One of the features of H.264 that contributes to such a high efficiency is the advanced motion estimation and compensation model. In our proposed video coder, all the advanced features of the motion model are preserved, including the multi block shapes, higher motion vector accuracy and multiple reference frames.

The encoding process of a frame is carried out in macroblocks (MB) in H.264. In our proposed system from [13], each MB is collected and rearranged into a displace frame difference (DFD) image, which is then fed into the MP coder. The proposed MP coder first performs a two-scale wavelet transforms over the DFD frame, and then applies the recursive atom finding and repairing procedure, using our distinctive code book. Finally, the atoms are quantized with a PLQ scheme and encoded with the MERGE coder to produce a fully embedded bit stream. In [13], we applied an atom limitation value, MaxAtom, in the proposed video coder to justify the bit allocation amongst frames. This system combines the advantages of the H.264 motion estimation and compensation model with the BUMP residual frame coder. Experiments show that good results can be achieved, especially at low and medium bit rates.

In this paper, we perform rate estimation analysis on the coding results and performance of this hybrid video system and present a model of the bit rate changes during the encoding process.

IV. RATE AND DISTORTION ESTIMATION FOR THE BUMP CODER

A. Overview

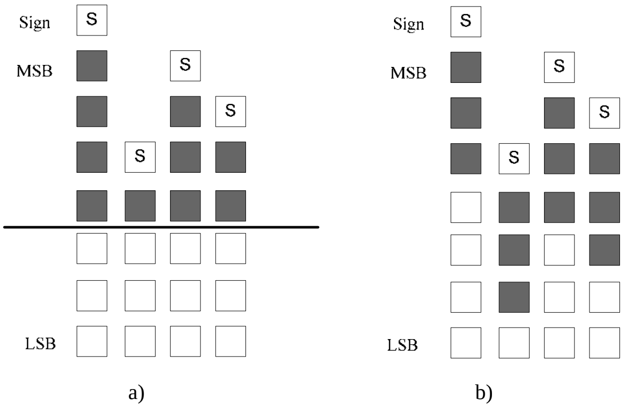
The BUMP codec encodes atoms in a special manner and therefore its rate distortion characteristics need particular study. The general encoding procedure of an MP scheme is to recursively search for the best approximation of the image using one basis function chosen from the codebook each time. At each iteration, the identity of the chosen basis function, its position in the image and the inner product value result. Such a result is defined as one atom [2]. In BUMP encoder, the coding can be briefly described as a three-step per atom process,

```
Initialize Compute a full set of inner products between
           the image and all bases in a codebook.
Repeat
    1. Find an atom. Full 1D or 2D search or reduced
       complexity strategies are possible.
    2. Image Update. Subtract quantized atom from
       image.
    3. Repair Inner Products. Re-compute required
       inner products only in atom footprint.
Until distortion or bit rate criterion met.
```

In [2], the found atoms are quantized, and then encoded in a raster scanning order according to their positions. The quantized atom value, the atom position and the basis indices are encoded respectively with adaptive Huffman code. Since BUMP applies totally different quantization and encoding methods to the atoms, the characteristics of its final rate-distortion curve are very different from those of previous MP codecs.

In BUMP, a combined PLQ and MERGE method is applied for encoding atoms. PLQ was first introduced in [14] as a new quantization scheme in order to achieve visually lossless compression result based on psycho-visual experiments. Later, it was found in [15] to fit well into MP-based visual compression algorithm to achieve a better objective quality. PLQ differs from traditional quantization schemes in the way it allocates quantization steps. In normal schemes it is common to divide a group of coefficients with one constant value, which is equivalent to truncating coefficients at the same bit, as show in Figure 3.a). Alternatively, in PLQ all the coefficients are quantized to the same number of refinement bits beyond their most significant bit (MSB), see Figure 3.b). PLQ shows a simple texture masking so that the large coefficients are quantized more coarsely than those with smaller values.

MERGE is a Recursive Group Embedded Coding algorithm. As introduced earlier, each atom chosen by BUMP is defined by a position (*XPos, YPos*) in the 2D data space with the magnitude A and 2D codebook basis index (B_x, B_y). To efficiently encode the atoms, PLQ is used as a top-down quantizer to quantize the magnitude of the atom.



a) b)

Figure 2. Comparison of two quantization schemes, a) Conventional quantization. b) PLQ quantization.

With PLQ, the magnitude of an atom's amplitude A is quantized into a triple parameter set $\langle S,\ F,\ R \rangle$, where S is the sign , F is the First Significant Bit (FSB) of A ($F = \log_2 |A|$) , and R is the remainder subject to the value of the PL in the range 0 to $2^{PL-1}-1$ ($PL>1$). The example illustrated in Figure 2 b) has a PL of 3.

Lossless coding of the attributes is done by the MERGE algorithm, in which atoms are gathered into groups with all attributes in common, and the positions are signaled by run length coding. This scheme works well with PLQ, which keeps the number of groups reasonably small. To reduce the number of groups further, the sign *S* of each atom is sent as one bit of side information which is efficient because positive and negative signs have near-equal probability. If the Precision Limit is *PL*, the MERGE algorithm is:

```
For F (the FSB) from Maximum to Minimum
  For R (the amplitude Residual) from ($2^{PL-1}$-1) to 0
    For each Basis Function K used
      Signal by Run Length Coding the position
       of each atom with attributes (F, R, K).
      Send the Sign S of the atom (1 bit)
    End of K (Basis Function) Group
  End of R (PLQ Residual) Group
End of F (FSB) Group
```

Maximum embedding is achieved by sending atoms in order of decreasing amplitude, with the codebook entry as the innermost loop. MERGE automatically compensates for variations in the frequency of occurrence of the attributes of the atoms and eliminates the need for entropy coding. The adaptive run length coding in MERGE adjusts to the statistics of the atom position.

B. Analysis of Rate-Atom relationship in BUMP

In a BUMP system, encoding is performed in an atom-wise manner. As the coding process continues, more and more atoms are coded and the number of used bits increases. It is therefore rational to look into the relationship between the bits rate and the number of encoded atoms. In later part of this paper, we call this the Rate-Atom (R-A) relationship.

Taking one frame from the test sequence Akiyo as an example, the coding process is displayed in detail in Table 1, which lists some lines quoted from a log file recording how the bits are used along the encoding process of a BUMP based coder.

According to the MERGE scheme, there are different kinds of information to be encoded for each frame. From Table 1 it can be seen that some bits are used for coding the file header and group information, such as group header, and some others for coding the random empty groups. Also, the number of bits used for each position depends on the atoms' spatial distribution, and is generally random.

Looking into the procedure of encoding a picture, the following factors contribute most to the ultimate bit rate:

Table 1. Example lines quoted from encoding log file.

Atom	Bits Cost	Total Bits Number	Information Transmitted
0	*29*	*29*	*header bits*
0	*8*	*37*	*FSBgroup_size*
0	*6*	*43*	*empty groups*
1	*17*	*60*	*atom sent*
1	*3*	*63*	*side information*
1	*1*	*64*	*sign coded*
2	*22*	*86*	*atom sent*
2	*3*	*89*	*side information*
2	*1*	*90*	*sign coded*
2	*7*	*97*	*End of Group*
2	*9*	*106*	*empty groups*
3	*20*	*126*	*atom sent*
3	*3*	*129*	*side information*
3	*1*	*130*	*sign coded*
3	*3*	*133*	*End of Group*
3	*4*	*137*	*empty groups*

i). the value of the amplitude of the atoms. Larger values take more bits to encoder;

ii). the distribution of data values. MERGE codes data by dividing it into groups, and a sparse data distribution means there are more groups involved, so that more bits are needed to deliver the group changes;

iii). the number of atoms to be encoded. Higher target quality often requires more atoms to be coded to transmit enough information to enable an image to be constructed with more fidelity.

A plot of the R-A relationship during coding this frame is shown in Figure 3. For each step in Table 1, there is one corresponding R-A point. Because various kinds of information needs to be encoded, the bit number increases a few times over a single atom. As indicated in Figure 3, such a phenomenon causes a high bit usage for atoms in the earlier stage, but has less impact to the later ones.

The exact number of steps for coding each atom depends on the data distribution, which might vary from frame to frame and is hard to track. Therefore, when discussing the R-A relationship in the following sections, we only consider the total bit cost after finishing coding each single atom.

In the next experiment, two test sequences, Akiyo and Foreman are encoded with the BUMP based video coder. Three frames of each are chosen and the Atom-Bit relationship is given in Figure 4. In this way, it can be seen that number of bits increases with the number of atoms and the plot shows a near linear relationship. To demonstrate such near-linear relationship, the standard linear functions are also shown in Figure 4. Based on observation, the bits number change can be roughly modeled with a straightforward linear function,

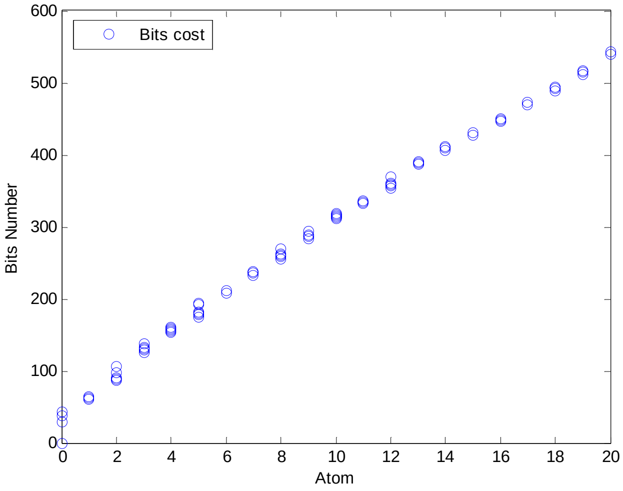


Figure 3. Bits increment during atom coding for a frame from Akiyo.

$$R = k \times A\,, \qquad (3)$$

where R is the bits number and A is the number of atoms that have been encoded.

This model from equation (3) has a very simple form, hence introducing a very low complexity in calculation when being applied to rate estimation, but also has an obvious disadvantage: it gives very limited accuracy. The actual bit changes are highly dependent on the frame content, and can show pretty large deviation from the linear line. As indicated in Figure 4, such inaccuracy is especially large over the earlier stage, where more bits are required for coding header information, and some later atoms, where the groups tend to have bigger sizes and fewer bits are need to deliver the atom information.

So, simply assuming a linear relationship is not sufficient to describe the trend of bits increments over the atoms. A more precise model is required to provide accurate bit rate estimation.

V. THE PROPOSED RATE-ATOM MODELLILNG FRAMEWORK

To find a better rate estimation model, further analysis needs to be carried out. Figure 5 shows the coding process of another inter frame taken from the test sequence Hall. This time, we connect the point representing the last atom to the origin. It is seen that the R-A curve lies above the straight line and is convex on the upper side, as in Figure 5. This means that the rate of increase in the number of bits reduces as the coding procedure progresses. It is shown in wider experimental results that this phenomenon remains valid for all the sample frames. A curve with such character can be modeled with a power function,

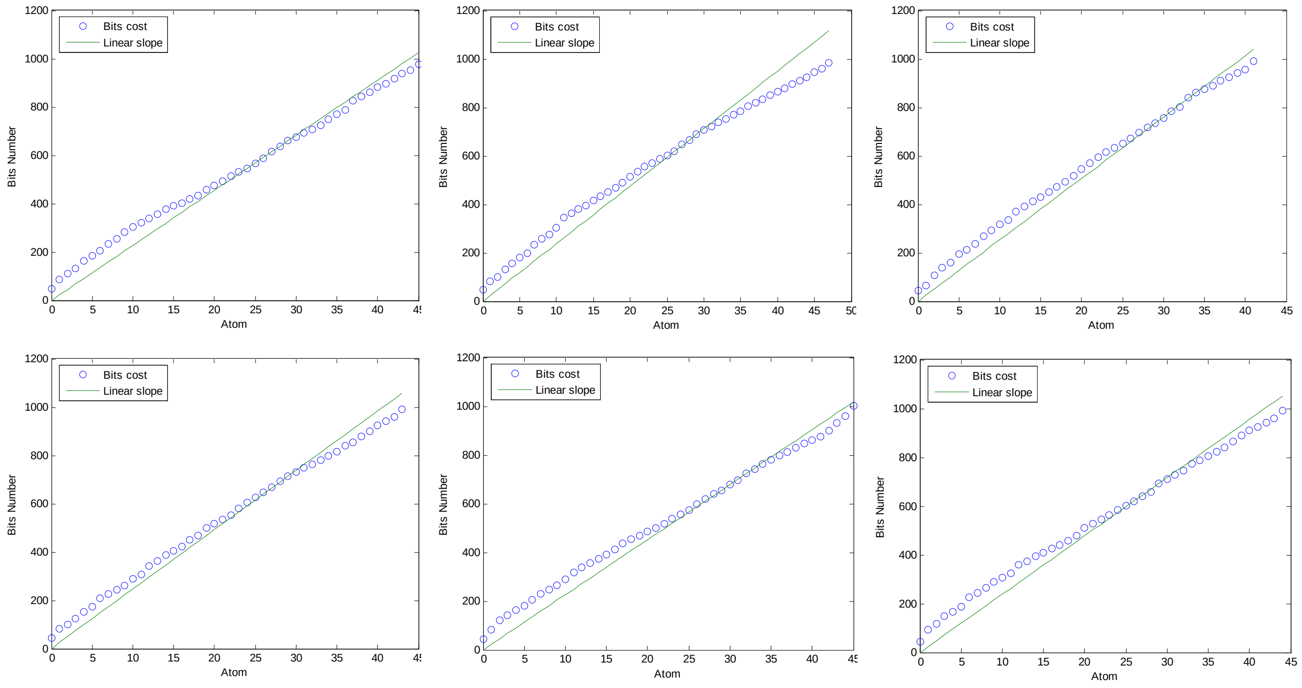


Figure 4. The Atom-Bits changes in sample frames from Akiyo (the first row) and Foreman(the second row).

$$f(x) = x^{\lambda}, \qquad \lambda < 1. \quad (4)$$

Considering the fact that the bit rate increases with a near linear character, the model describing the relationship between the atom number and bits rate can be developed as

$$R = \theta \times A^{\lambda} + b, \qquad (5)$$

This model in equation (5) comes in a very simple form. θ describes the general trend of the bit rate increment, λ defines how the curve deviates from the linear line, and b is introduced to cover some extra bits for encoding side information.

With a fixed λ value of 0.9, and a b value of 0, the model is applied to some sample frames and the results are given in Figure 6. It can be seen the model fits the estimated line along the real R-A curve very well, and therefore, this model has much potential to be used for estimating the bit rate changes along the BUMP encoding process.

The correlation coefficients of the modeled values and the actual bit rate are plotted in Figure 7, where eight frames are taken from each of the four sample sequences. The chosen sequences include those containing complex motion, such as Stefan, and those involving very little movement, such as Akiyo. It can be seen that across the various styles of video content the correlation coefficient is always larger than 0.99, which proves that the model is accurate and robust to variation in video content, and therefore is able to effectively capture the character of the R-A relationship.

VI. RATE ESTIMTION SCHEME WITH THE PROPOSD MODEL

The proposed framework accurately models the relationship between the coded atom numbers and the bits rate cost, and therefore is a powerful tool for operational rate estimation. In this section, we study how to set its parameters to achieve accurate rate estimation.

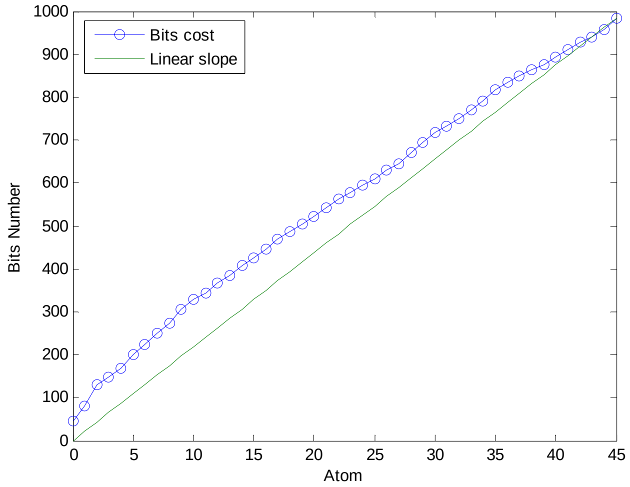


Figure 5. Deviation of the R-A curve, for coding a frame from Hall, from the standard linear plot.

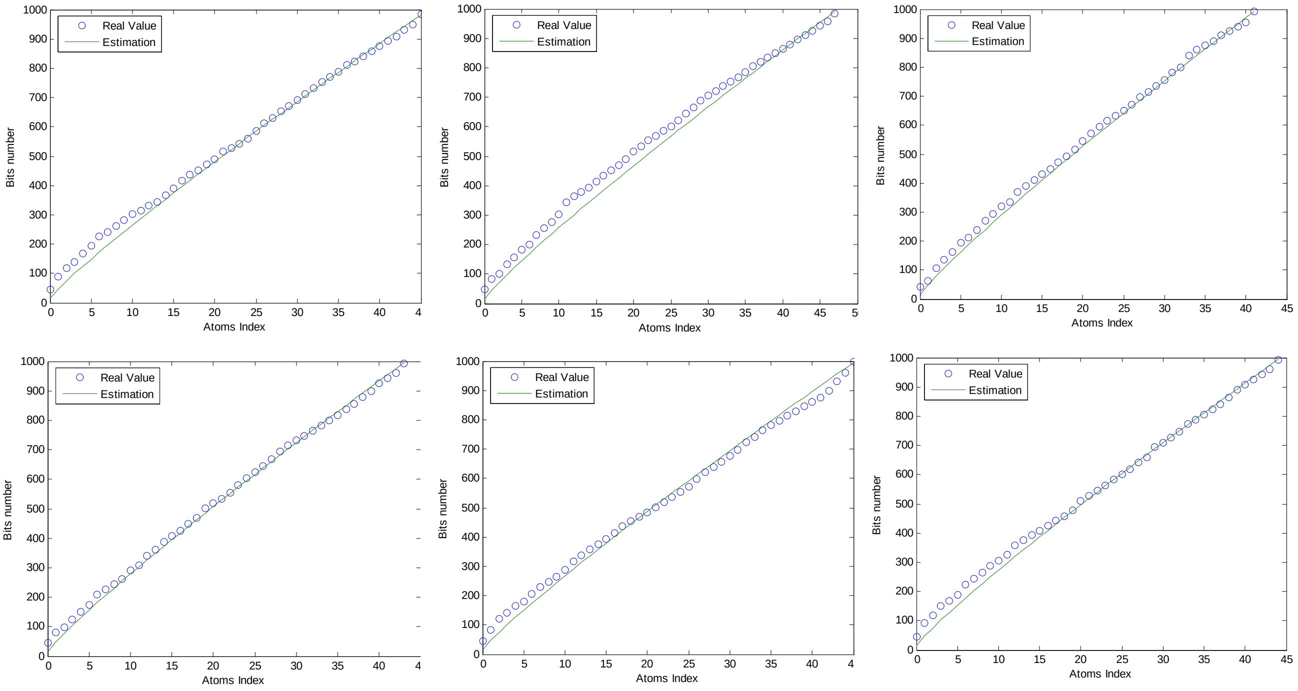


Figure 6. The modeled Atom-Bits relationship and original data of sample frames from Akiyo (the first row) and Foreman (the second row).

A. Updating parameters of the proposed framework for bit rate modelling

Once the framework function is defined, the accuracy of the estimation mainly relies on the choice of the parameter values. In practice, operational R-D modeling constructs the R-D curves by mathematical processing of the observed data [7]. The parameters of the framework are calculated based on some encoding results of the visual data for compression. As the coding process progresses, the model's parameters need to be updated.

The proposed framework comes in a very simple form, and there are only three parameters to be calculated. The parameter defines the shape of the R-D curve. We have mentioned that the value for must be smaller than 1 to coincide with the real R-A curve, which shows a reduced speed of increase of bit rate as the atom number increases. θ describes the general near linear trend of the bit rate increment, and can be seen as a linear regulation factor to direct the model. The two parameters interact with each other and therefore their values need to be determined together. In Figure 8 we plot the modeled curves obtained from the framework with the same θ value and various λ values. The curve with λ = 0.9 is a decent model for the rate of coding the sample frame. When λ varies, the resultant curves deviate significantly from the original one. θ is the regulator that ensures the slope of the curve matches that of the actual data. The experiments show that θ must be adjusted according to the value of λ.

To summarize, the model can be updated by calculating θ after the value of λ has been chosen. In this manner, the three curves in Figure 8 are all directed to

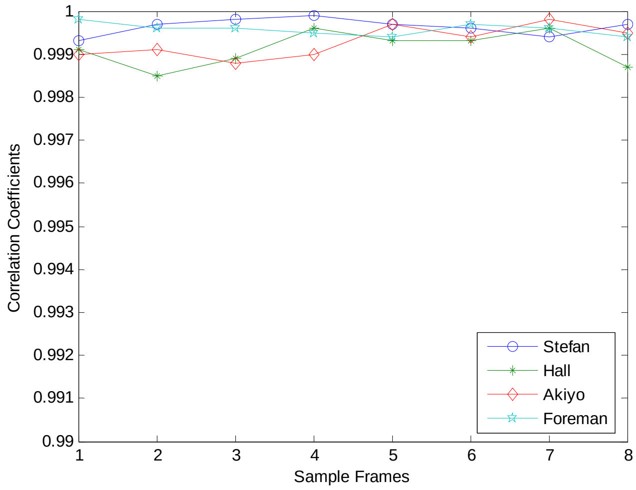


Figure 7. The correlation between the values from the model and the actual bit rate.

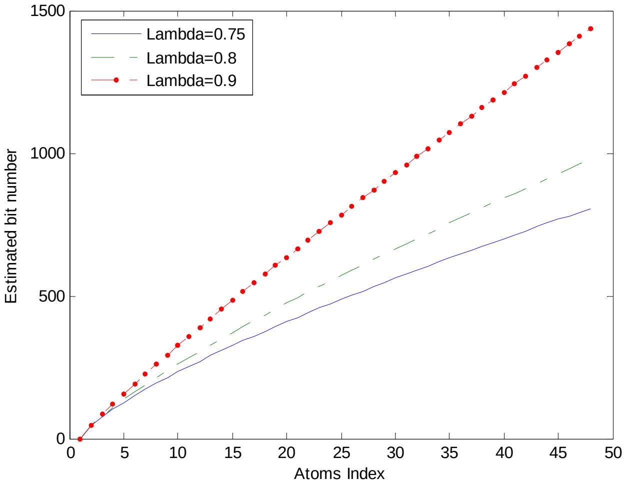


Figure 8. The modeled curves from the framework with same θ value and various λ values.

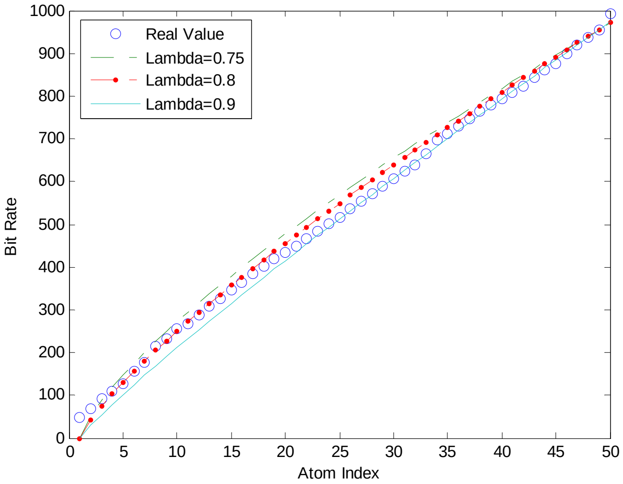


Figure 9. Rate estimations with different model parameter values.

follow the trend of the actual data, as shown in Figure 9.

B. A rate estimation scheme for BUMP coder based on the proposed R-A model

A closer study of Figure 9 indicates that the three modeled curves each simulate the actual data well for certain specific ranges while showing increased errors over others. This indicates that the curve shape varies over the encoding procedure, and any chosen curve shape is only valid within certain local ranges. Therefore, applying an adaptive parameter updating scheme which considers only the recently coded atoms might be able to track the shape changes more closely.

Considering all the above characteristics of the visual signals and the framework, we propose the following bit rate estimation algorithm for BUMP based encoders,

Step 1). Initialize the model. Set a local “window” size of atom numbers. Carry out the BUMP frame encoding and keep the resultant bit rate until one “window” of atoms has been encoded. Keep the resultant bit number of those atoms for Step 2. In our experiment, the “window” size is set as 4.

Step 2). Update the model. Use the obtained bit number of the recently coded “window” size of atoms to select the best value of λ from a range of candidate values. The range for λ is set to [0.7, 0.95] in our experiments.

Step 3). Calculate the θ value for the framework with the estimated λ value, and previous bit number.

Step 4). Estimate the bit rate after coding the next atom with the two parameters above. If the atom to be estimated is the last one in that frame, the parameter b must be introduced into the model, with an empirical value of b_0=15. The estimated bit rate R is therefore,

$$R = \theta \times A^{\lambda} + b_0,, \quad \left(\begin{array}{ll} b_0 = 15 & last\_atom \\ b_0 = 0 & else \end{array} \right) \quad \textbf{(6)}$$

Step 5). Encode the next atom, record the resultant bit rate, and go back to Step 2).

The range of λ in Step 2 is defined according to a large number of experiment results. The R-A slope always shows a near linear relationship, and therefore the value of λ is always close to 1.

When the encoding process meets the criterion, either the target rate or quality, and needs to be truncated, on top of those already used for coding the atoms, some more bits are required to encode the End of Group, and End of File signs, which makes it necessary to introduce the parameter of *b*. The exact required number of bits varies, and an average value of 15 is used.

VII. EXPERIMENTAL RESULTS

The bit rate estimation scheme presented in section VI is implemented and applied in the hybrid video coder with BUMP and H.264, which has been described in section III. To assess the accuracy of the proposed scheme, four video test sequences, namely Akiyo, Hall, Foreman and Stefan, are encoded with frame types being set as IPPPP. The first I frame is encoded with H.264 and all successive P frame are encoded with the BUMP inter frame coder. The rate estimation scheme is performed over the inter frames. For each frame, the estimation is recorded at each stage, and eventually plotted to be compared with the actual bits cost.

Since the “windowed” data collection focuses the parameter estimation on the recent coded atoms, it closely follows the actual changes of the R-A slope. Plotted in Figure 10 are the estimated and the actual number of bits for encoding the 5^{th} frame from the Foreman sequence. It can be clearly seen that the model efficiently detects the changes in the R-A curve’s shape and gives a precise prediction. Figure 11 shows the relative error (RE) of the estimation, which is defined as

$$RE = \frac{Estimation\_Error}{Actual\_Bits} \times 100\% . \qquad \textbf{(7)}$$

Since the model does not start estimating until 4 atoms have been encoded, we only plot the relative error from the 5th atom onwards. The estimation error quickly converges to near zero and, from the 13th atom onwards, remains within the range of ±0.02.

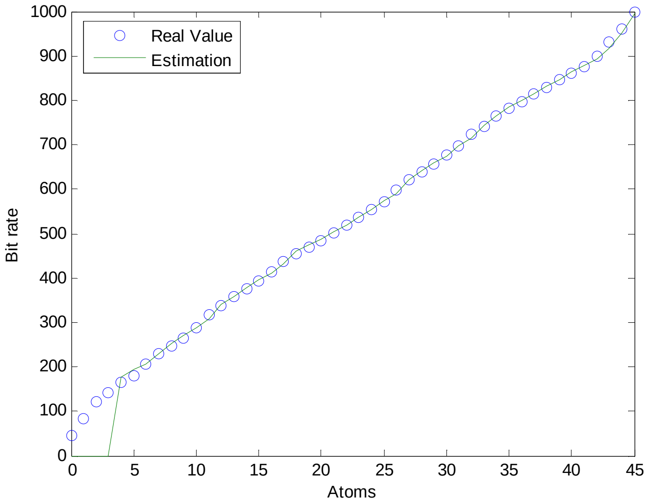


Figure 10. Comparison of bit estimation and the actual bit number for frame 5 in Foreman.

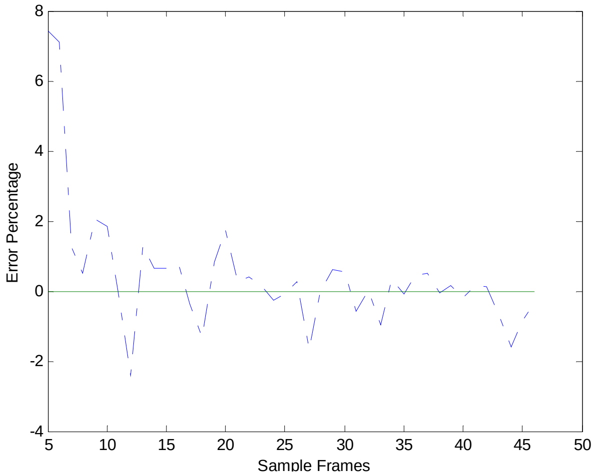


Figure 11. The relative estimation error for frame 5 of Foreman.

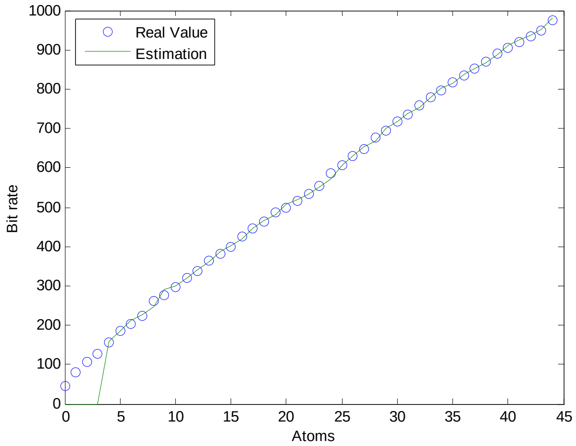


Figure 12. Comparison of bit estimation and the actual bit number for frame 8 in Stefan.

Similar results are obtained from coding Stefan. The results for coding the 8th frame of Stefan are plotted in Figure 12 and Figure 13, where the algorithm also shows a very high accuracy. Although the relative error is nearly 6% at the 10th atom, it soon becomes smaller than 2%.

The average values of more experimental results are given in Table 2. To ensure the accuracy in averaging, both Mean Error (ME) and Relative Error (RE) are calculated in absolute values, as defined in equations **(8)** and **(9)**,

$$ME = \frac{\sum_{Estimations} \left|Estimation\_Error\right|}{Estimations} \qquad (8)$$

$$RE' = \frac{\sum_{Estimations} \dfrac{\left|Estimation\_Error\right|}{Actual\_Bits} \times 100\%}{Estimations} \qquad (9)$$

where *Estimations* refers to the total number of estimated values within this frame. From the data in the table, most RE values are smaller than 1.5%, which indicates a very high prediction accuracy.

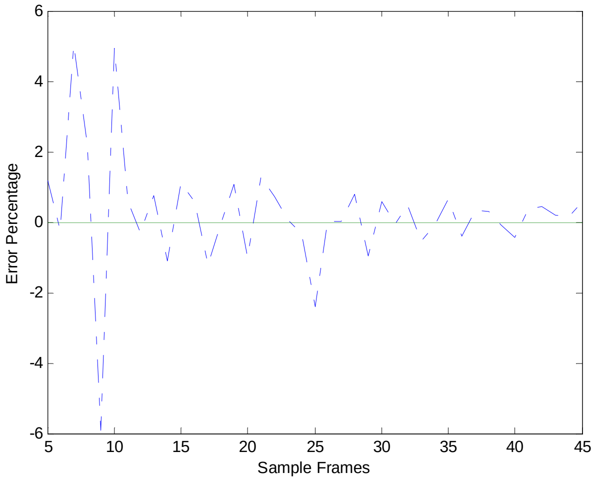


Figure 13. The relative estimation error for frame 8 of Stefan.

VIII. CONCLUSION AND FUTURE WORK

We have proposed a bit rate modeling framework for the Bath University Matching Pursuit video coder. A detailed theoretical justification for the unique property of the R-A curves in the BUMP encoder is provided. Based on this modeling framework, a rate estimation algorithm has been proposed. Experiments show that the proposed algorithm effectively tracks the variations of the R-A curve during the encoding procedure, providing a high precision rate control mechanism. There are two major contributions of this work. First, we present a R-A analysis methodology for a MP coding scheme which is a novel approach with a very low complexity and extremely high accuracy. The second contribution is the efficient rate estimation algorithm which is robust to video contents variations and able to effectively track the localized R-A curve shape changes.

The mechanism of MP allows a straightforward distortion measure at encoding stage. Together with the powerful rate estimation framework presented above, a Rate-Distortion optimization scheme for BUMP is to be developed in the next stage of this project.

ACKNOWLEDGMENT

The authors wish to thank Wei Huo and Xiaopeng Wang for their contributions to the BUMP coding system.

REFERENCES

[1] S. G. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries”, *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[2] R. Neff and A. Zakhor, “Very low bit rate video coding based on matching pursuits”, *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 7, pp. 158–171, 1997.

[3] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard", *IEEE Trans. Circuits and Systems for Video Tech*, vol. 13, pp. 560-576. 2003.

[4] Y. Yuan and D. M. Monro, “Improved Matching Pursuits Image Coding”,. *Proceedings. IEEE Int. Conf. Acoustics, Speech, and Signal Processing,* pp.201-204, 2005.

Table 2. Estimation errors from encoding four sequences. ME refers to the Mean estimation Error, and RE refers to he Relative estimation Error.

Akiyo		Foreman		Hall		Stefan	
ME	RE(%)	ME	RE(%)	ME	RE(%)	ME	RE(%)
3.7361	*0.8435*	4.1021	*1.0343*	3.5220	*1.0070*	4.0141	*0.8887*
4.4163	*1.0863*	4.1422	*0.8534*	4.0210	*0.8851*	3.1203	*0.7893*
3.8639	*1.0023*	4.0148	*1.0046*	4.3148	*1.1369*	4.1084	*1.0870*
4.3485	*1.0437*	3.2557	*0.8449*	3.2250	*0.9555*	3.8585	*1.0901*
4.5462	*1.0399*	3.8750	*0.9899*	4.2818	*1.2454*	3.2926	*0.8596*
4.4115	*1.1256*	3.8693	*1.0590*	3.5384	*0.9084*	3.8965	*0.9831*
3.6539	*0.8851*	4.8866	*1.3028*	4.4915	*1.1269*	4.2381	*1.0668*
4.6096	*1.1297*	3.5869	*0.8668*	3.4139	*0.9264*	3.9780	*0.9626*

[5] Y. Yuan, A.N. Evans and D.M. Monro, “Low complexity Separable Matching Pursuits”, *IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp.725-728, Montreal, May 2004.

[6] D. M. Monro, “Basis Picking for Matching Pursuits Image Coding”, *IEEE Int. Conf. Image Processing*, pp.2495-2498 Singapore, October 2004.

[7] Zhihai He and Sanjita K. Mitra, “From Rate-Distortion analysis to Resource-Distortion analysis,” *IEEE Circuits and System Magzine*, pp. 6-18, Third Quarter 2005.

[8] J. Ribas-Corbera and S. Lei, “Rate Control in DCT Video Coding for Low-Delay Communication,” *IEEE Trans. Circuits and Systems for Video Tech*, vol. 9, no.1, pp. 172-185, Feb. 1999.

[9] T. Chiang and Y.-Q. Zhang, “A new rate control scheme using quadratic rate distortion model,” *IEEE Trans. Circuits and Systems for Video Tech*, vol. 7, pp. 246–250, Feb. 1997.

[10] W. Ding and B. Liu, “Rate control of MPEG video coding and recording by rate-quantization modeling,” *IEEE Trans. Circuits System Video Tech*, vol. 6, pp. 12–20, Feb. 1996.

[11] Z. He and S.K. Mitra, “A unified rate-distortion analysis framework for transform coding,” *IEEE Trans. Circuits and Systems for Video Tech*, vol.11, pp. 1221–1236, Dec. 2001.

[12] R. Neff and A. Zakhor, “Matching-Pursuit Video Coding-Part II: Operational Models for Rate and Distortion”, *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 12, pp. 27–39, 2002.

[13] Haoxiang Zhang, X. Wang, W. Huo and D.M.Monro, “A Hybrid Video Coder Based on H.264 with Matching Pursuits”, *IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp.889-892,Toulouse, May 2006.

[14] D. M. Monro, J-L Aufranc, M. A.Bowers and W Poh, “Visual embedding of wavelet transform coefficients”, *IEEE Int. Conf. Image Process.*, pp.186-189,Sept. 2000.

[15] W. Poh and D.M. Monro, “Comparison of residual compression methods in motion compensated video*”, IEEE Int. Workshop on Multimedia Signal Processing*, pp.109-112, Virgin Islands, Dec. 2002.

[16] Karsten Sühring, H.264/AVC Reference Software, http://iphome.hhi.de/suehring/tml/

Haoxiang Zhang was born in Hebei China, in 1978. He received the BEng degree in Electronic Engineering from Hunan University, Changsha, China, in 2000. He is now working towards his PhD in the University of Bath, Bath, UK, specializing in Rate Control in Digital Video Coding. His main research interests include video and image compression, signal and image processing, and data compression. He is a student member of IEEE.

Adrian N. Evans received the B.Eng. degree from Loughborough University of Technology, Loughborough, U.K., in 1990, and the Ph.D. degree in medical image processing from the University of Southampton, Southampton, U.K., in 1994. From 1993 to 1994, he was with the University of York, U.K., as a Postdoctoral Research Associate. From 1994 to 1997, he was a Lecturer in Information Engineering at Massey University, Palmerston North, New Zealand. Since September 1997, he has been a Lecturer in the Department of Electronic and Electrical Engineering, University of Bath, U.K. His current research interests include motion estimation, nonlinear filtering, colour image analysis and video coding.

Donald M. Monro received the BASc (1964) and MASc (1966) degrees from the University of Toronto. He spent a brief period in the aeronautical industry before studying for the PhD degree at Imperial College, London, which was awarded in 1971, by which time he was a lecturer in electrical engineering. He was subsequently a senior lecturer in electrical engineering from 1979 and in the Department of Computing from 1983, both at Imperial College, before taking up the chair of electronics at the University of Bath in 1991. He is an emeritus professor in electronic and electrical engineering at the University of Bath, UK. In addition to video compression, his research interests are in biometrics, and more general aspects of signal processing. He is a member of the IEEE.