1-D Transforms for the Motion Compensation Residual

1-D Transforms for the Motion Compensation Residual

Fatih Kamisli, *Student Member, IEEE,* and Jae S. Lim, *Fellow, IEEE*

***Abstract*—Transforms used in image coding are also commonly used to compress prediction residuals in video coding. Prediction residuals have different spatial characteristics from images, and it is useful to develop transforms that are adapted to prediction residuals. In this paper, we explore the differences between the characteristics of images and motion compensated prediction residuals by analyzing their local anisotropic characteristics and develop transforms adapted to the local anisotropic characteristics of these residuals. The analysis indicates that many regions of motion compensated prediction residuals have 1-D anisotropic characteristics and we propose to use 1-D directional transforms for these regions. We present experimental results with one example set of such transforms within the H.264/AVC codec and the results indicate that the proposed transforms can improve the compression efficiency of motion compensated prediction residuals over conventional transforms.**

***Index Terms*—Discrete cosine transforms, Motion compensation, Video coding**

I. Introduction

AN important component of image and video compression systems is a transform. A transform is used to transform image intensities. A transform is also used to transform prediction residuals of image intensities, such as the motion compensation (MC) residual, the resolution enhancement residual in scalable video coding, or the intra prediction residual in H.264/AVC. Typically, the same transform is used to transform both image intensities and prediction residuals. For example, the 2-D Discrete Cosine Transform (2-D DCT) is used to compress image intensities in the JPEG standard and MC-residuals in many video coding standards. Another example is the 2-D Discrete Wavelet Transform (2-D DWT), which is used to compress images in the JPEG2000 standard and high-pass prediction residual frames in inter-frame wavelet coding [1]. However, prediction residuals have different spatial characteristics from image intensities [2], [3], [4], [5]. It is of interest therefore to study if transforms better than those used for image intensities can be developed for prediction residuals.

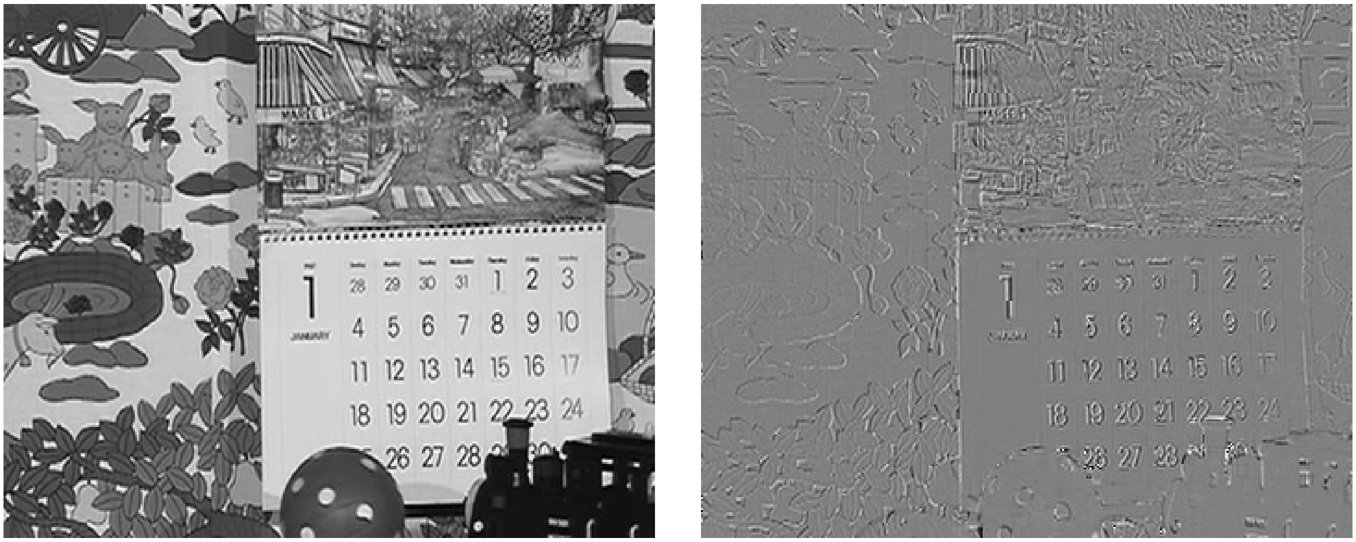
Recently, new transforms have been developed that can take advantage of locally anisotropic features in images [6], [7], [8], [9], [10]. A conventional transform, such as the 2-D DCT or the 2-D DWT, is carried out as a separable transform by cascading two 1-D transforms in the vertical and horizontal dimensions. This approach favors horizontal or vertical features over others and does not take advantage of locally anisotropic features present in images. For example, the 2-D DWT has vanishing moments only in the horizontal and vertical directions. The new transforms adapt to locally anisotropic features in images by performing the filtering along the direction where image intensity variations are smaller. This is achieved by resampling the image intensities along such directions [7], by performing filtering and subsampling on oriented sublattices of the sampling grid [9], by directional lifting implementations of the DWT [10], or by various other means. Even though most of the work is based on the DWT, similar ideas have been applied to DCT-based image compression [8].

In video coding, prediction residuals of image intensities are coded in addition to image intensities. Many transforms have been developed to take advantage of local anisotropic features in images. However, investigation of local anisotropic features in prediction residuals has received little attention. Inspection of prediction residuals shows that locally anisotropic features are also present in prediction residuals. Unlike in image intensities, a large number of pixels in prediction residuals have negligibly small amplitudes. Pixels with large amplitudes concentrate in regions which are difficult to predict. For example, in motion compensation residuals, such regions are moving object boundaries, edges, or highly detailed texture regions. Therefore a major portion of the signal in MC residuals concentrates along such object boundaries and edges, forming 1-D structures along them. Such structures can be easily seen in Figure 1. As a result, in many regions anisotropic features in MC residuals typically manifest themselves as locally 1-D structures at various orientations. This is in contrast to image intensities, which have 2-D anisotropic structures.

In this paper, we present block transforms specifically designed for MC residuals. We first analyze the difference between images and MC residuals using both visual inspection and an adaptive auto-covariance characterization. This analysis reveals some differences between images and MC residuals. In particular, it shows how locally anisotropic features in images appear in MC residuals. Based on this analysis, we propose new transforms for MC residuals. We then show potential gains achievable with a sample set of such transforms using the reference software of H.264/AVC.

The remainder of the paper is organized as follows. In Section II, differing characteristics of images and MC residuals are discussed and analyzed. Then a sample set of block transforms is introduced in Section III. Section IV discusses various aspects of a system implementation with these transforms. Experimental results with the reference software of H.264/AVC are then presented in Section V, and the paper is concluded in Section VI.

Authors are with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 02139 USA e-mail: fkamisli@mit.edu, jslim@mit.edu



(a) Image

(b) Motion compensation (MC) residual

Fig. 1. Frame 10 of mobile sequence at CIF resolution and its MC-residual predicted from frame 9 using ful-pel motion estimation with 8x8-pixel blocks.

II. Analysis of motion compensation residuals

This section first presents an empirical analysis of characteristics of images and motion compensated prediction residuals based on visual inspection using the image and its MC residual shown in Figure 1, and then provides an auto-covariance analysis that quantifies the discussed differences.

A common aspect of MC residuals is that smooth regions can be predicted quite well. For example, the prediction residuals of uniform background regions in Figure 1(b) are negligibly small. The spatial correlation in smooth regions of images is high and this enables successful prediction. In motion compensated prediction, even if the underlying motion is not exactly translational, the high spatial correlation of pixels enables a quite accurate match between blocks in such regions. In texture regions, prediction does not work as well as in smooth regions. For example, in Figure 1(b) the calendar picture on the top right corner contains many fine details and prediction in this region does not work well. Even though the local variations in such regions can not be predicted well, the local mean can be predicted well and the local mean of prediction residuals in such regions is typically zero.

Prediction also does not work well around object boundaries or edges. Consider the boundary of the ball and the boundary of the objects in the background, or the edges of the letters on the calendar in Figure 1. In all these regions, the boundaries or edges contain large prediction errors in the residual frame. In motion compensated prediction, motion is typically not exactly translational and this results in a mismatch along an edge or boundary and produces large prediction errors along these structures.

Characteristics of images and MC residuals differ significantly around object boundaries or edges. It is the rapidly changing pixels along the boundary or edge of the original image that can not be predicted well and large prediction errors form along these structures in MC residuals. These structures are 1-D structures and the residuals concentrating on these structures have 1-D characteristics. Such 1-D structures can be easily seen in the MC residual in Figure 1(b). Boundary or edge regions in images, on the other hand, have typically smooth structures on either side of the boundary or edge and their characteristics are 2-D.

Prior statistical characterizations of MC residuals focused on representing its auto-covariance with functions that provide a close fit to experimental data using one global model for the entire MC residual [4], [5], [3]. To show the differences of local anisotropic characteristics in images and MC residuals, we use two models for the auto-covariance of local regions. One is a separable model and the other generalizes it by allowing the axes to rotate. We estimate the parameters of these models from images and MC residuals and plot the estimated parameters. These plots provide valuable insights.

A. Auto-covariance models

A stationary Markov-1 signal has an auto-covariance given by equation (1).

$$R(I) = \rho^{|I|} \tag{1}$$

For discrete-time stationary Markov-1 signals, the decorrelating transform can be obtained analytically [11] and this transform becomes the well-known DCT as correlation reaches its maximum ($\rho \to 1$.) A 2-D auto-covariance function formed from equation (1) using separable construction is given by equation (2).

$$R_s(I,J) = \rho_1^{|I|} \rho_2^{|J|} \tag{2}$$

Due to separable construction, the decorrelating transform for this auto-covariance is the 2-D DCT (as $\rho_1 \to 1$, $\rho_2 \to 1$.) The good performance of the 2-D DCT in image compression is due to high correlation of neighboring pixels in images and $\rho_1 = \rho_2 = 0.95$ has been considered a good approximation for typical images [11].

The separable model in equation (2) has also been used to characterize the MC residual and it has been reported that

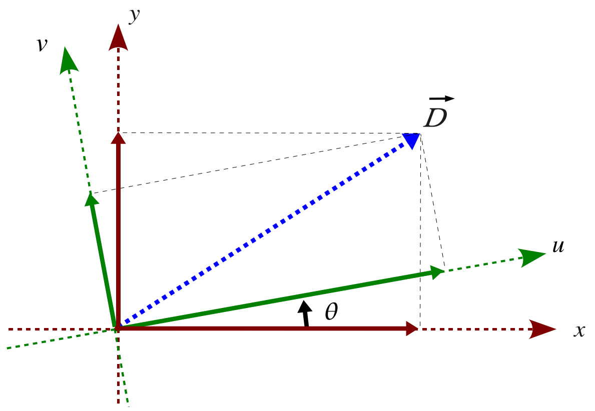


Fig. 2. Comparison of separable and the generalized auto-covariance models. Use of the separable model corresponds to expanding the distance vector D in the cartesian coordinate system. Use of the generalized model corresponds to expanding the distance vector $\vec{D}$ in a rotated coordinate system.

the correlations are weaker than in images. Other models have been proposed to model the weaker correlations more precisely [4], [5]. These models are global and were proposed to provide a closer fit to the average auto-covariance of the MC residual obtained from different parts of a frame. All these models are global and separable, and cannot adequately capture local anisotropies in images and MC residuals.

To capture local anisotropies in images and MC residuals, we use a generalized model, shown in equation (3).

$$R_g(\theta, I, J) = \rho_1^{|Icos(\theta)+Jsin(\theta)|}\rho_2^{|-Isin(\theta)+Jcos(\theta)|} \quad (3)$$

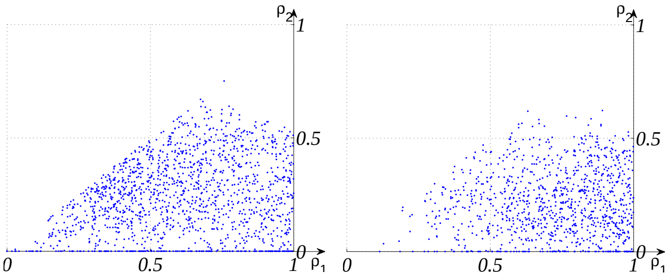
This model has an additional degree of freedom provided by the parameter θ . The parameter θ allows rotation of the axes of the auto-covariance model and enables capturing local anisotropic features by adjusting to these features. The separable model is a special case of the generalized model. The generalized model with $\theta = 0°$ is the separable model. Figure 2 shows both models. Characterization of images with similar generalized auto-covariance models have been made [10]. Characterizations of images and MC residuals with the separable model, or its derivatives, have also been made [11], [4], [5], [3]. However, MC residuals have not been characterized with a direction-adaptive model.

B. Estimation of parameters of auto-covariance models

We estimate the parameters ρ_1 and ρ_2 for the separable model, and the parameters ρ_1, ρ_2 and θ for the generalized model from blocks of 8x8-pixels of the image and the MC residual shown in Figure 1. We first use the unbiased estimator to estimate a non-parametric auto-covariance of each block. This is accomplished by removing the mean of the block, correlating the zero mean-block with itself, and dividing each element of the correlation sequence by the number of overlapping points used in the computation of that element. Then we find the parameters ρ_1, ρ_2 and θ so that the models in equations (2) and (3) best approximate the estimated non-parametric auto-covariance, by minimizing the mean-square-error between the non-parametric auto-covariance estimate and the models. In the minimization, we use lags less than four (i.e. $|I|, |J| < 4$) because at large lags the number of overlapping points becomes less and the estimates become noisy. We use ρ_1 for the larger covariance coefficient and let θ vary between $0°$ and $180°$. The estimation results are shown in Figure 3 for the image and in Figure 4 for the MC residual. Each point in the figures represents the estimate from one 8x8-pixel block.

C. Estimated model parameters for images

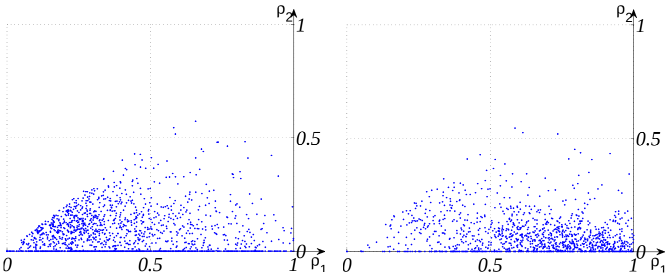
First, consider the scatter plots shown in Figures 3(a) and 3(b). They were obtained from the image shown in Figure 1(a). In the plot from the separable model (Figure 3(a)), the points fill most regions, except the northeast corner where both ρ_1 and ρ_2 are large. This indicates that the parameters ρ_1 and ρ_2 have large variability when modeled with the separable model. In the plot from the generalized model (Figure 3(b)), the points tend to concentrate in the southeast corner where ρ_1 is typically larger than 0.5 and ρ_2 smaller than 0.5. Significantly fewer points have a ρ_1 less than 0.5 compared to the separable case. This has two implications. First, the variability of parameters ρ_1 and ρ_2 of the auto-covariance is reduced, when modeled with the generalized model. Reduction of variability is important as it can model the source better and may lead to better compression of the source. Second, ρ_1 is typically larger than 0.5 and this means the generalized model can often capture high correlation from the source. The parameter θ adjusts itself such that ρ_1 points along directions with smaller variations than in the separable model. This is



(a) Separable model

(b) Generalized model

Fig. 3. Scatter plots of (ρ_1,ρ_2)-tuples estimated using the separable and generalized auto-covariance models from the image shown in Figure 1.



(a) Separable model

(b) Generalized model

Fig. 4. Scatter plots of (ρ_1,ρ_2)-tuples estimated using the separable and generalized auto-covariance models from the MC residual shown in Figure 1.

consistent with the resampling and lifting methods in [7] and [10], which perform filtering along directions with smaller variations than the predefined horizontal or vertical directions.

D. Estimated model parameters for MC residuals

We consider the scatter plots obtained from the MC residual shown in Figure 4(a) and 4(b). The plot obtained using the separable model (Figure 4(a)) has typically a ρ_1 smaller than 0.5. This is in contrast to the typical ρ_1 in Figure 3(a) which is larger than 0.5. MC residuals usually are more random since they are the parts of images which could not be predicted well, and ρ_1 tends to be smaller.

Even though MC residuals are more random than images, many regions of MC residuals still have some structure. The separable model can not capture those well and produces a small ρ_1 estimate. Figure 4(b) shows the estimated ρ_1 and ρ_2 when the auto-covariance of the MC residual is modeled with the generalized model. In this case, many more points have a ρ_1 larger than 0.5 compared to the separable case (Figure 4(a)). The majority of the points have a large ρ_1 and a small ρ_2.

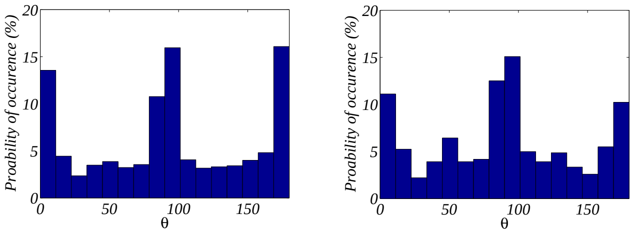
In summary, if the auto-covariance of MC residuals is modeled with the separable model, estimated ρ_1 (and ρ_2) are both typically small. If the generalized model is used, then typically ρ_1 is large and ρ_2 is small. An estimated large ρ_1 indicates that some structure has been captured from the local region in the MC residual. The combination of a large ρ_1 and a small ρ_2 indicates that the structure exists only along the direction of the ρ_1, indicating a 1-D structure.

E. Comparison of estimated model parameters for images and MC residuals

Figures 3 and 4 also illustrate the difference of the locally anisotropic features between the image and the MC residual. Consider the generalized auto-covariance characterization of the image and the MC residual in Figures 3(b) and 4(b). In both plots, the majority of the points have a ρ_1 larger than 0.5. However, the points in the plot of the MC residual have a smaller ρ_2. In other words, given any (ρ_1, ρ_2)-tuple in the image characterization, the smaller covariance factor becomes even smaller in the MC residual characterization. This is a major difference in the statistical characteristics between images and the MC residuals.

F. Estimated angles (θ) using the generalized model

We also provide plots of the estimated angles (θ) of the generalized auto-covariance model from the image and the MC residual shown in Figure 1. The plots are shown in Figure 5. The highest peaks in the plots are at around $0°$, $90°$ and $180°$, where peaks at $0°$ and $180°$ correspond to horizontally aligned features, and a peak at $90°$ corresponds to vertically aligned features. This indicates that the image and MC residual shown in Figure 1 have more horizontal and vertical features than features along other directions.



(a) Image

(b) MC-residual

Fig. 5. Histograms of estimated angles (θ) of the generalized auto-covariance model from the image and MC residual in Figure 1.

III. 1-D DIRECTIONAL TRANSFORMS

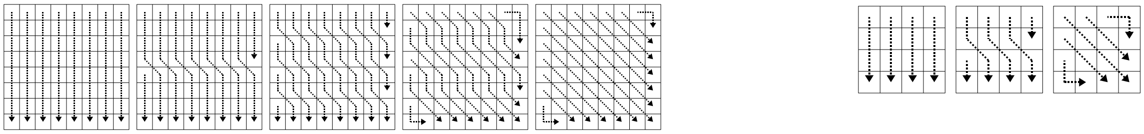
Based on visual inspection of MC residuals and the results of the auto-covariance characterization in Section II, a large number of local regions in MC residuals consist of 1-D structures, which follow object boundaries or edges present in the original image. This indicates that using 2-D transforms with basis functions that have 2-D support may not be the best choice for such regions. We propose to use transforms with basis functions whose support follow the 1-D structures of MC residuals. Specifically, we propose to use 1-D directional transforms for MC residuals.

Since we compress MC residuals using the H.264/AVC codec in our experiments, we discuss sets of 1-D directional transforms, specifically 1-D directional DCT's, on 8x8-pixel and 4x4-pixel blocks. We note that the idea of 1-D transforms for prediction residuals can also be extended to wavelet transforms [12].

The 1-D directional transforms that we use in our experiments are shown in Figure 6. We use sixteen 1-D block transforms on 8x8-pixel blocks and eight 1-D block transforms on 4x4-pixel blocks. Figure 6(a) shows the first five 1-D block transforms defined on 8x8-pixel blocks. The remaining eleven are symmetric versions of these five and can be easily derived. Figure 6(b) shows the first three 1-D block transforms defined on 4x4-pixel blocks. The remaining five are symmetric versions of these three and can be easily derived.

Each of the 1-D block transforms consists of a number of 1-D patterns which are all directed at roughly the same angle, which would correspond to the direction of the large covariance coefficient. For example, all 1-D patterns in the fifth 1-D block transform defined on 8x8-pixel blocks or the third 1-D block transform defined on 4x4-pixel blocks are directed towards south-east. The angle is different for each of the 1-D block transforms and together they cover $180°$, for both 8x8-pixel blocks and 4x4-pixel blocks. Each 1-D pattern in any 1-D block transform is shown with arrows in Figure 6 and defines a group of pixels over which a 1-D DCT is performed. We note that these 1-D patterns have different lengths and do not extend to neighboring blocks, creating block transforms that can be applied on a block-by-block basis.

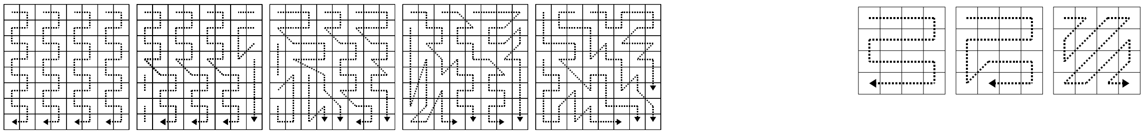
Even though 1-D directional transforms improve the compression of MC residuals for many regions, the 2-D DCT is essential. There exist regions in MC residuals which can be better approximated with 2-D transforms. Therefore, in our experiments, we use both 1-D directional transforms and the



(a) First five out of sixteen 1-D transforms are shown. Each arrow indicates a 1-D DCT on the tranversed pixels. Remaining eleven transforms are symmetric versions.

(b) First three out of eight 1-D transforms are shown. Each arrow indicates a 1-D DCT on the tranversed pixels. Remaining five transforms are symmetric versions.

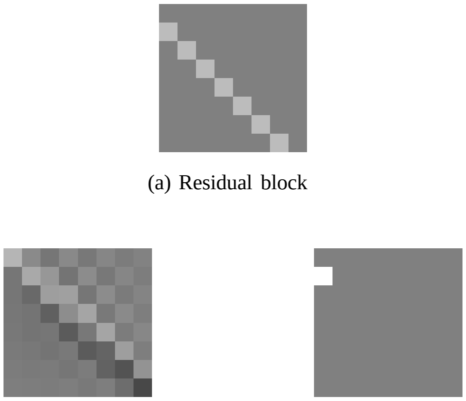
Fig. 6. 1-D directional transforms defined on (a) 8x8-pixel blocks and (b) 4x4-pixel blocks. Each transform consists of a number of 1-D DCT's defined on groups of pixels shown with arrows.



(a) Scans for 1-D transforms shown in Figure 6(a).

(b) Scans for 1-D transforms shown in Figure 6(b).

Fig. 7. Scans used in coding the quantized coefficients of 1-D transform defined on (a) 8x8-pixel blocks and (b) 4x4-pixel blocks.

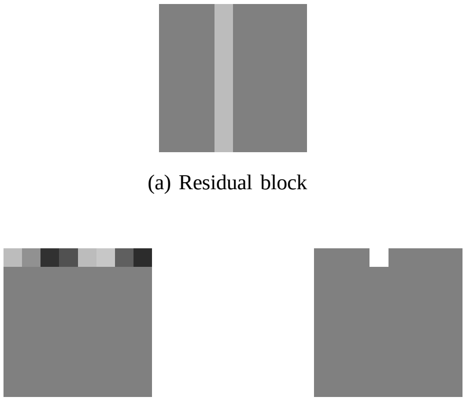


(a) Residual block

(b) Transform coefficients obtained with 2-D DCT

(c) Transform coefficients obtained with 1-D Transform

Fig. 8. Comparison of 2-D DCT and 1-D directional transform on an artificial residual block consisting of a diagonal 1-D structure (mid-gray level represents zero). To represent the residual block, 2-D DCT requires many nonzero transforms coefficients while the 1-D transform requires only one nonzero transform coefficient.



(a) Residual block

(b) Transform coefficients obtained with 2-D DCT

(c) Transform coefficients obtained with 1-D Transform

Fig. 9. Comparison of 2-D DCT and 1-D directional transform on an artificial residual block consisting of a vertical 1-D structure (mid-gray level represents zero). To represent the residual block, 2-D DCT requires many nonzero transforms coefficients while the 1-D transform requires only one nonzero transform coefficient.

2-D DCT. Encoders with 1-D transforms have access to 2-D DCT and can adaptively choose to use one among the available 1-D transforms and the 2-D DCT.

To show the effectiveness of the proposed transforms we present two examples in Figures 8 and 9. Figure 8(a) shows a sample residual block, Figure 8(b) shows the transform coefficients obtained by transforming the block with the 2-D DCT, and Figure 8(c) shows the transform coefficients obtained by transforming the block with a 1-D transform aligned with the structure in the residual (the specific transform used is 1-D Transform #5 in Figure 6(a)). The mid-gray level in these figures represents zero, and the residual block consists of an artificially created 1-D structure aligned diagonally. Such a residual block can possibly be obtained from the prediction of a local region which contains a diagonal edge separating two smooth regions in the original image block. To represent this residual block, 2-D DCT requires many nonzero transform coefficients while the 1-D transform requires only one nonzero transform coefficient.

The second example is shown in Figure 9. The residual block in this example consists of a vertical 1-D structure. Figure 9(c) shows the transform coefficients obtained by transforming the block with a 1-D transform aligned with the vertical structure in the residual (the specific transform used is 1-D Transform #1 in Figure 6(a)), and this block can be represented with a single nonzero transform coefficient. The transform coefficients obtained by transforming the block with the 2-D DCT are shown in Figure 9(b). We note that the separable 2-D DCT can be performed by first applying 1-D transforms along the vertical dimension and then applying 1-D transforms along the horizontal dimension. The first set of horizontal 1-D transforms is equivalent to the 1-D transform used in Figure 9(c). As a result, when performing the separable 2-D DCT, the result of the first set of vertical 1-D transforms provides already a good representation of the block (since only a single nonzero coefficient suffices, as shown in Figure 9(c)), and applying the second set of horizontal 1-D transforms results in more nonzero coefficients. In summary, for residual blocks with a 1-D structure, even if the alignment of the structure is consistent with the directions of the 2-D transform, 1-D transforms can represent such blocks better.

IV. Integration of 1-D transforms into the H.264/AVC codec

To integrate the proposed 1-D transforms into a codec, a number of related aspects need to be carefully designed. These include the implementation of the transforms, quantization of the transform coefficients, coding of the quantized coefficients, and coding of the side information which indicates the selected transform for each block. The overall increase of complexity of the codec is also an important aspect in practical implementations.

In H.264/AVC, transform and quantization are merged together so that both of these steps can be implemented with integer arithmetic using addition, subtraction and bitshift operations. This has many advantages including the reduction of computational complexity [13]. In this paper, we use floating point operations for these steps for simplicity. This does not change the results. We note that it is possible to merge the transform and quantization steps of our proposed 1-D transforms so that these steps can also be implemented with integer arithmetic.

A. Coding of 1-D transform coefficients

Depending on the chosen entropy coding mode in H.264/AVC, the quantized transform coefficients can be encoded using either context-adaptive variable-length codes (CAVLC mode) or context-adaptive binary arithmetic coding (CABAC mode). In both cases, coding methods are adapted to the characteristics of the coefficients of the 2-D DCT. Ideally, it would be best to design new methods which are adapted to the characteristics of the coefficients of the proposed 1-D transforms. For the experiments in this paper, however, we use the method in H.264/AVC in CAVLC mode with the exception of the scan. We use different scans for each of the 1-D transforms.

Figure 7(b) shows the scans for the 1-D transforms defined on 4x4-pixel blocks shown in Figure 6(b). These scans were designed heuristically so that coefficients less likely to be quantized to zero are closer to the beginning of the scan and coefficients more likely to be quantized to zero are closer to the end of the scan. Scans for the remaining 1-D transforms defined on 4x4 blocks are symmetric versions of those in Figure 7(b).

For transforms defined on 8x8-pixel blocks, H.264/AVC generates four length-16 scans instead of one length-64 scan, when entropy coding is performed in CAVLC mode. Figure 7(a) shows the four length-16 scans for each of the 1-D transforms defined on 8x8-pixel blocks shown in Figure 6(a). These scans were designed based on two considerations. One is to place coefficients less likely to be quantized to zero closer to the beginning of the scan and coefficients more likely to be quantized to zero closer to the end of the scan. The other consideration is to group neighboring 1-D patterns into one scan. The 1-D structures in prediction residuals are typically concentrated in one region of the 8x8-pixel block and the 1-D transform coefficients representing them will therefore be concentrated in a few neighboring 1-D patterns. Hence, grouping neighboring 1-D patterns into one scan enables capturing those 1-D transform coefficients in as few scans as possible. More scans that consist of all zero coefficients can lead to more efficient overall coding of coefficients.

B. Coding of side information

The identity of the selected transform for each block needs to be transmitted to the decoder so that the decoder can use the correct inverse transform for each block. We refer to this information as side information. In this paper, we use a simple procedure to code the side information.

If a macroblock uses 8x8-pixel transforms, then for each 8x8-pixel block, the 2-D DCT is represented with a 1-bit codeword, and each of the sixteen 1-D transforms is represented with a 5-bit codeword. If a macroblock uses 4x4-pixel transforms, then for each 4x4-pixel block, the 2-D DCT can be represented with a 1-bit codeword and each of the eight 1-D transforms can be represented with a 4-bit codeword. Alternatively, four 4x4-pixel blocks within a single 8x8-pixel block can be forced to use the same transform, which allows us to represent the selected transforms for these four 4x4-pixel blocks with a single 4-bit codeword. This reduces the average bitrate for the side information but will also reduce the flexibility of transform choices for 4x4-pixel blocks. We use this alternative method that forces the use of the same transform within an 8x8-pixel block in our experiments because it usually gives slightly better results.

We note that the simple method that we used in this paper can be improved by designing codewords that exploit the probabilities of the selected transforms.

C. Complexity increase of codec

Having a number of transforms to choose from increases the complexity of the codec. An important consideration is the increase in encoding time. This increase depends on many factors of the implementation and can therefore vary considerably. Our discussion of the increase in encoding time is based only on the reference software of H.264/AVC in high complexity encoding mode.

In high-complexity encoding mode, RD (Rate Distortion) optimized encoding is performed, where each available coding option for a macroblock or smaller blocks is encoded and the option(s) with the smallest RD-cost is chosen. The implementation within the reference software is designed for general purpose processors and executes each command successively, with no parallel processing support. Therefore, each coding option is encoded successively. Within each coding option, each block is encoded with each available transform. Hence, the amount of time spent on transform (T), quantization (Q), entropy coding of quantized coefficients (E), inverse quantization (Q), and inverse transform (T) computations increases linearly with the number of available transforms. The factor of increase would be equal to the number of transforms if the computation of the additional transforms (and inverse transforms) takes the same amount of time as the conventional transform. Because the conventional transform is 2-D while our proposed transforms are 1-D, the factor of increase can be represented with αN_{tr}, where N_{tr} is the number of transforms and α is a scaling constant less than 1. The increase of the overall encoding time is typically equal to the increase in TQEQT computation time because other relevant computations, such as computing the RD-cost of each transform, are negligible.

The TQEQT computation time is a fraction of the overall encoding time. In our experiments on P-frames with 8x8-block transforms, about 30% of the encoding time is used on TQEQT computations with the conventional transform. The increase in encoding time is a factor of 5.8 (=$17\alpha 30\% + 70\%$ where $\alpha = 1$). The actual increase is expected to be significantly less than 5.8 with a more accurate choice of α and integer-point implementations of transform computations.

The decoding time does not increase. The decoder still uses only one transform for each block, which is the transform that was selected and signaled by the encoder. In fact, the decoding time can decrease slightly because the decoder now uses 1-D transforms for some blocks and 1-D transforms require less computations than the 2-D DCT.

V. Experimental Results

We present experimental results to illustrate the compression performance of the proposed 1-D directional transforms on motion compensation (MC) residuals using the H.264/AVC codec (JM reference software 10.2). We compare the compression performance of the proposed transforms with that of the conventional transform (2-D DCT.) We also study the effect of block sizes for the transforms. Hence, each encoder in our experiments has access to a different set of transforms which may vary in size and in type. The available sizes are 4x4 and/or 8x8. The available types are $2Ddct$ (2-D DCT) or $1D$ (1-D directional transforms.) Note that encoders with $1D$ type transforms have access to the conventional transform, as discussed in Section III. As a result, we have the following encoders.

- 4x4-2Ddct
- 4x4-1D (includes 4x4-2Ddct)
- 8x8-2Ddct
- 8x8-1D (includes 8x8-2Ddct)
- 4x4-and-8x8-2Ddct
- 4x4-and-8x8-1D (includes 4x4 and 8x8-2Ddct)

Some detail of the experimental setup is as follows. We use 11 QCIF (176x144) resolution sequences at 30 frames-per-second (fps), 4 CIF (352x288) resolution sequences at 30 fps, and one 720p (1280x720) resolution sequence at 60 fps. All sequences are encoded at four different picture quality levels (with quantization parameters 24, 28, 32 and 36), which roughly corresponds to a range of 30dB to 40dB in PSNR. Entropy coding is performed with context-adaptive variable length codes (CAVLC). Rate-distortion (RD) optimization is performed in high-complexity mode. In this mode, all possible macroblock coding options are encoded and the best option is chosen. Selection of the best transform for each block is also performed with RD optimization by encoding each block with every available transform and choosing the transform with the smallest RD cost.

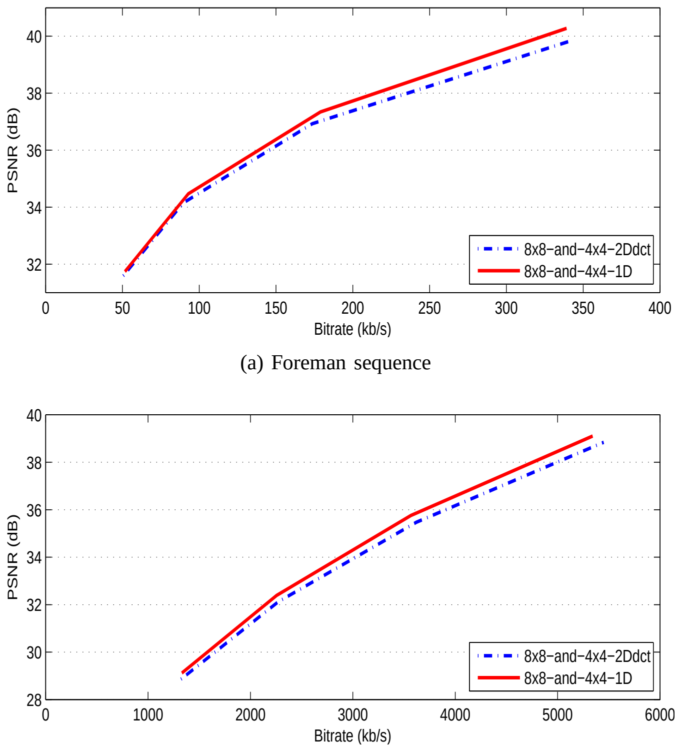
We encode the first 20 frames for the 720p sequence and the first 180 frames for all other sequences. The first frame is encoded as an I-frame, and all remaining frames are encoded as P-frames. Since these experiments focus on the MC residual, intra macroblocks use always the 2-D DCT and inter macroblocks choose one of the available transforms for each block. Motion estimation is performed with quarter-pixel accuracy and the full-search algorithm using all available block-sizes.

We evaluate encoding results with bitrate (in kbit/sec) and PSNR (in dB). The bitrate includes all encoded information including transform coefficients from luminance and chrominance components, motion vectors, side information for chosen transforms, and all necessary syntax elements and control information. The PSNR, however, is computed from only the luminance component. The proposed transforms are used only for the luminance component, and coding of chrominance components remains unchanged.

A. *Rate-Distortion plots*

We first present experimental results with Rate-Distortion curves for two sequences. Figure 10 shows Bitrate-PSNR plots for Foreman (QCIF resolution) and Basket (CIF resolution) sequences. The results are provided for two encoders which have both access to 4x4 and 8x8 sizes but different types of transforms. It can be observed that 4x4-and-8x8-1D has better compression performance at all encoding bitrates.

It can also be observed that the (horizontal or vertical) separation between the 4x4-and-8x8-2Ddct and 4x4-and-8x8-1D plots increases with increasing picture quality. This typically translates to a higher PSNR improvement at higher picture qualities. It also implies a higher percentage bitrate saving at higher picture qualities for many sequences. For example, the PSNR improvement is 0.1dB at 75kb/s and 0.47dB at 325kb/s for the Foreman sequence. Similarly, the percentage bitrate savings are 2.24% at 32dB and 8.15% at 39dB. The increase of separation between the plots is in part because at higher picture qualities, the fraction of the total bitrate used to code the transform coefficients of the MC residual data is larger than at lower picture qualities. For example, for the Foreman sequence, about 30% of the entire bitrate is used to code the transform coefficients of the MC residual data at low picture qualities and 55% at high picture qualities. The lower the fraction is, the lower will be the impact of improved compression efficiency through the use of $1D$ transforms on the overall bitrate saving. An additional factor that increases the separation between Bitrate-PSNR plots at higher picture qualities is the transmitted side information that indicates the chosen transforms. At lower picture qualities, the side information requires a higher fraction of the entire bitrate and becomes a larger burden.



(a) Foreman sequence

(b) Basket sequence

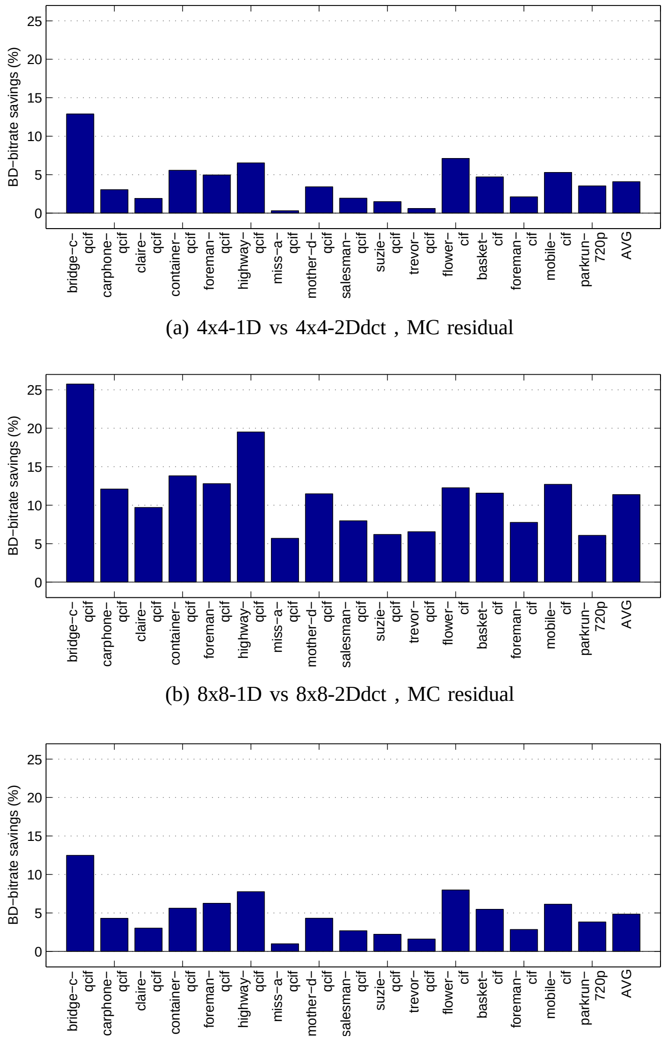
Fig. 10. Bitrate-PSNR plots for Foreman (QCIF) and Basket (CIF) sequences.

B. Bjontegaard-Delta bitrate results

To present experimental results for a large number of sequences we use the Bjontegaard-Delta (BD) bitrate metric [14]. This metric measures the average horizontal distance between two Bitrate-PSNR plots, giving the average bitrate saving over a range of picture qualities of one encoder with respect to another encoder. Using the BD-bitrate metric, the comparisons of encoders with access to $1D$ transforms with encoders with access to $2Ddct$ transform(s) is shown in Figure 11. Figure 11(a) compares 4x4-1D to 4x4-2Ddct, Figure 11(b) compares 8x8-1D to 8x8-2Ddct, and Figure 11(c) compares 4x4-and-8x8-1D to 4x4-and-8x8-2Ddct. The average bitrate savings are 4.1%, 11.4% and 4.8% in each of Figures 11(a), 11(b) and 11(c).

Bitrate savings depend on the block size of the transforms, which is typically also the block size for prediction. Bitrate savings are largest when encoders which have access to only 8x8-pixel block transforms are compared and smallest when encoders which have access to only 4x4-pixel block transforms are compared. This is in part because the distinction between 2-D transforms and 1-D transforms becomes less when block-size is reduced. For example, for 2x2-pixel blocks, the distinction would be even less, and for the extreme case of 1x1-pixel blocks, there would be no difference at all.

The results also show that the bitrate savings depend on the characteristics of the video sequences. The ranking in performance among different sequences tends to remain unchanged among the three cases. The $bridge - c - qcif$ sequence has the largest savings and the $miss - a - qcif$ sequence has the smallest savings in Figures 11(a), 11(b) and 11(c).



(a) 4x4-1D vs 4x4-2Ddct , MC residual

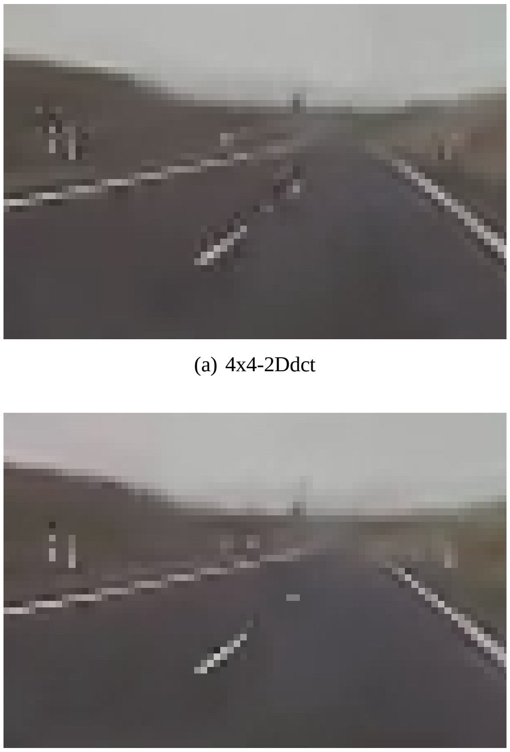
(b) 8x8-1D vs 8x8-2Ddct , MC residual

(c) 4x4-and-8x8-1D vs 4x4-and-8x8-2Ddct , MC residual

Fig. 11. Average bitrate savings (using BD-bitrate metric [14]) of several encoders with access to $1D$ transforms with respect to encoders with only conventional transform(s). Each plot provides savings when different sized transforms are available.

C. Visual quality

Video sequences coded with 1-D transforms have in general better overall visual quality. Although the improvements are not obvious, they are visible in some regions in the reconstructed frames. Regions with better visual quality typically include sharp edges or object boundaries. Figure 12 compares a portion of the reconstructed frame 101 of highway sequence (QCIF) coded with 4x4-2Ddct and 4x4-1D at 19.90 kb/s and 20.43 kb/s, respectively. The stripes on the road are cleaner and the poles on the sides of the road are sharper in the frame reconstructed with 4x4-1D. Figure 13 compares a portion of the reconstructed frame 91 of basket sequence (CIF) coded with 8x8-2Ddct and 8x8-1D at 1438 kb/s and 1407 kb/s, respectively. The shoulders and faces of the players are cleaner in the frame reconstructed with 8x8-1D.



(a) 4x4-2Ddct

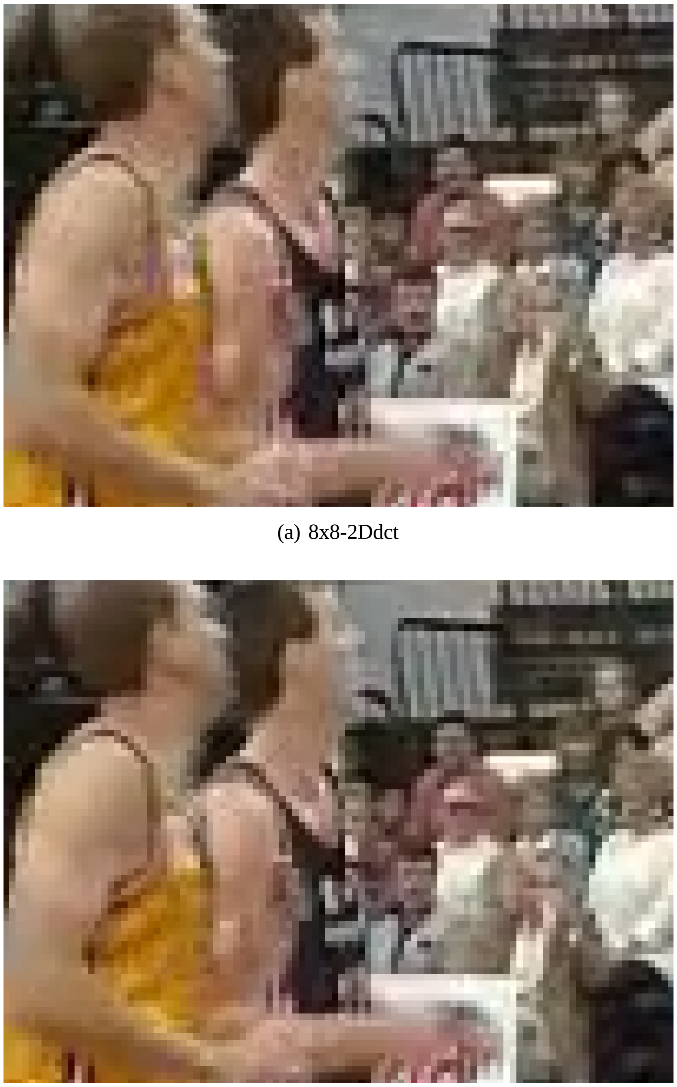
(b) 4x4-1D

Fig. 12. Comparison of a portion of the reconstructed frame 101 of highway sequence (QCIF) coded with 4x4-2Ddct and 4x4-1D at 19.90 kb/s and 20.43 kb/s, respectively. Frame 101 was coded at 33.117 dB PSNR using 680 bits with the 4x4-2Ddct and at 33.317 dB PSNR using 632 bits with the 4x4-1D.

D. Bitrate for coding side information

The encoder sends side information to indicate the chosen transform for each block. The side information can be a significant fraction of the overall bitrate. Figure 14 shows the average percentage of the bitrate used to code the side information in the 4x4-and-8x8-1D encoder for each sequence. These numbers are averages obtained from encoding results at all picture quality levels using quantization parameters 24, 28, 32 and 36. The average percentage bitrate used to code the side information is 4.4%.

We note that the percentage of the bitrate used to code the side information for each individual sequence in Figure 14(a) correlates with the average bitrate savings of that sequence shown in Figure 11(c). For example, $miss-a-qcif$ sequence has the smallest bitrate savings in Figure 11(c), and the smallest percentage bitrate to code the side information in Figure 14. In general, if sequence A has larger bitrate savings than sequence B, then sequence A also has a larger percentage bitrate for the side information. This is because bitrate savings typically happen when the prediction residuals of the sequence have more $1D$ structures. This means more frequent use of $1D$ transforms relative to *2-D DCT*, which in turn implies a larger bitrate for the side information.



(a) 8x8-2Ddct

(b) 8x8-1D

Fig. 13. Comparison of a portion of the reconstructed frame 91 of basket sequence (CIF) coded with 8x8-2Ddct and 8x8-1D at 1438 kb/s and 1407 kb/s, respectively. Frame 91 was coded at 28.834 dB PSNR using 49360 bits with the 8x8-2Ddct and at 29.166 dB PSNR using 47632 bits with the 8x8-1D.

The average percentages of bitrate used to code the side information for different encoders are as follows. Among the encoders with access to $1D$ transforms, the average percentages are 3.6% for 4x4-1D, 5.9% for 8x8-1D and 4.4% for 4x4-and-8x8-1D. These are averages obtained from all sequences at all picture qualities. The lowest fraction is used by 4x4-1D and the highest fraction is used by 8x8-1D. The 4x4-1D uses a 1-bit (2-D DCT) or a 4-bit (1-D transforms) codeword for every four 4x4-pixel blocks with coded coefficients, and the 8x8-1D uses a 1-bit or a 5-bit codeword for every 8x8-pixel block with coded coefficients. In addition, the probability of using a 1-D transform is higher in 8x8-1D than in 4x4-1D.

E. Probabilities for selection of transforms

How often each transform is selected is presented in Figure 15. Probabilities obtained from all sequences for the 4x4-and-8x8-1D encoder are shown in Figure 15(a) for low picture qualities and in Figure 15(b) for high picture qualities. It can be observed that the 2-D DCT's are chosen more often than the other transforms. A closer inspection reveals that using a 1-bit codeword to represent the 2-D DCT and a 4-bit codeword

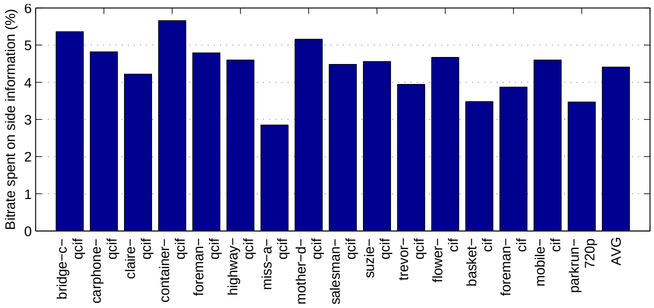
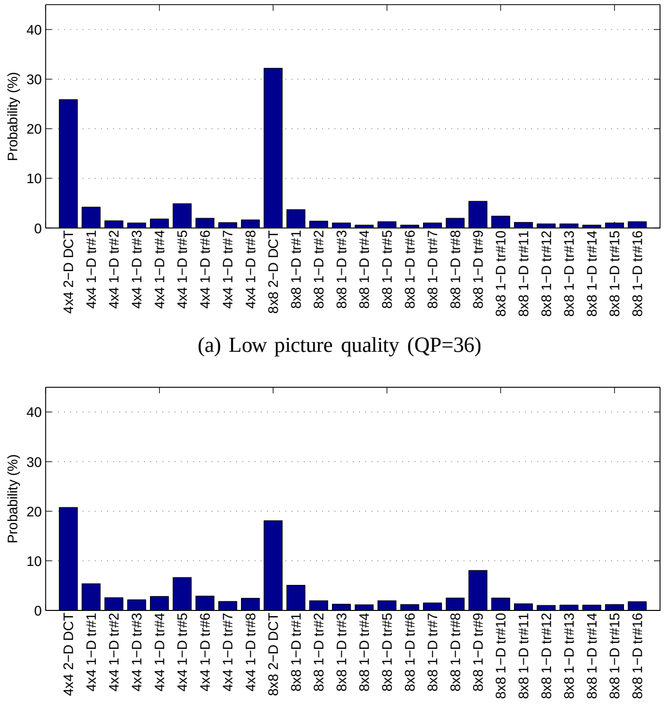


Fig. 14. Average percentages of total bitrate used to code side information of 4x4-and-8x8-1D for all sequences. Numbers are obtained from all encoded picture qualities.

(5-bit in case of 8x8-pixel transforms) to represent the 1-D transforms is consistent with the numbers presented in these figures.



(a) Low picture quality (QP=36)

(b) High picture quality (QP=24)

Fig. 15. Average probability of selection for each transform at different picture quality levels for 4x4-and-8x8-1D.

At low picture qualities, the probability of selection is 58% for both 2-D DCT's, and 42% for all 1-D transforms. At high picture qualities, the probabilities are 38% for both 2-D DCT's, and 62% for all 1-D transforms. The 1-D transforms are chosen more often at higher picture qualities. Choosing the 2-D DCT costs 1-bit, and any of the 1-D transforms 4-bits (5-bits for 8x8-pixel block transforms). This is a smaller cost for 1-D transforms at high bitrates relative to the available bitrate.

Note that the 2-D DCT is the most often selected transform, but when all 1-D transforms are combined, the selection probabilities of the 2-D DCT and all 1-D transforms are roughly equal. This means that a 1-D transform is chosen as often as a 2-D transform for a given block of the MC residual.

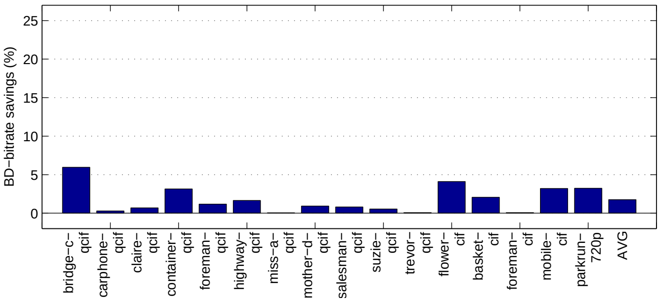


Fig. 16. Average bitrate savings of an encoder with access to $2D$ directional transforms [8] with respect to an encoder with only conventional 2-D DCT transforms for MC residuals. Specifically, 4x4-and-8x8-2D vs 4x4-and-8x8-2Ddct.

F. *Comparison with 2-D Directional Transforms*

In this section, we compare a specific directional block transform proposed for image compression with our 1-D transforms on MC residuals. These directional block transforms, proposed by Zeng et al. [8] are 2-D directional DCT's together with a DC separation and ΔDC correction method borrowed from the shape-adaptive DCT framework in [15].

We present experimental results with these transforms from [8]. These transforms are 2-D directional block transforms designed to exploit local anisotropic features in images. It is typical to use transforms that are originally developed for image compression, to compress prediction residuals. Our intent here is to provide experimental evidence indicating that although 2-D directional transforms can improve compression efficiency for images [8], they are worse than 1-D transforms for improving compression efficiency of MC residuals.

For the experiments, we have complemented the six transforms in [8] with another eight transforms to achieve finer directional adaptivity (which is comparable to the adaptivity of our proposed transforms) in case of 8x8-pixel block transforms. For 4x4-pixel block transforms, we designed six transforms using the techniques provided in [8]. The scanning patterns for the transform coefficients were also taken from [8] and coding of the chosen transform is done similar to the coding of the proposed 1-D directional transforms.

We compare an encoder with $2D$ directional transforms (including 2-D DCT) to an encoder with $2Ddct$ transforms in Figure 16. Specifically, we compare 4x4-and-8x8-2D directional transforms with 4x4-and-8x8-2Ddct on MC residuals. The average bitrate saving is 1.8%, which is lower than the average saving obtained with $1D$ transforms in Figure 11(c), which was 4.8%.

VI. Conclusions and Future Work

In this paper, we proposed 1-D directional transforms for the compression of motion compensation (MC) residuals. MC residuals have different spatial characteristics from images. Both signals have locally anisotropic features, but their characteristics are different. Unlike in images, local regions in MC residuals have many pixels with amplitudes close to zero.

Pixels with large amplitudes concentrate in regions which are difficult to predict, such as moving object boundaries, edges, or highly detailed texture regions, and form 1-D structures along them. Hence a significant portion of anisotropic features in MC residuals have 1-D characteristics, suggesting the use of 1-D transforms for such regions. Experimental results using a sample set of such transforms within the H.264/AVC codec illustrated the potential improvements in compression efficiency. Gains depend on the characteristics of the video and on the block size used for prediction.

In our experiments, we did not design coefficient coding methods that are adapted to the characteristics of coefficients of the proposed transforms. Instead, we changed only the scanning pattern of transform coefficients and the remaining coding methods were not modified. These methods are adapted to the characteristics of the conventional transform. Characteristics of coefficients of the proposed transforms can be different and adapting to these characteristics can improve the overall compression efficiency. Another area for future research is to investigate potential gains achievable with the proposed transforms in compressing other prediction residuals such as the intra prediction residual in H.264/AVC, resolution enhancement residual in scalable video coding, or the disparity compensation residual in multi view video coding.

References

[1] J. Ohm, M. v. Schaar, and J. W. Woods, "Interframe wavelet coding - motion picture representation for universal scalability," *EURASIP Signal Processing: Image Communication, Special Issue on Digital Cinema*, vol. 19, pp. 877–908, October 2004.

[2] F. Kamisli and J. Lim, "Transforms for the motion compensation residual," *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 789–792, April 2009.

[3] K.-C. Hui and W.-C. Siu, "Extended analysis of motion-compensated frame difference for block-based motion prediction error," *Image Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 1232–1245, May 2007.

[4] C.-F. Chen and K. Pang, "The optimal transform of motion-compensated frame difference images in a hybrid coder," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 40, no. 6, pp. 393–397, Jun 1993.

[5] W. Niehsen and M. Brunig, "Covariance analysis of motion-compensated frame differences," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 9, no. 4, pp. 536–539, Jun 1999.

[6] W. Ding, F. Wu, and S. Li, "Lifting-based wavelet transform with directionally spatial prediction," *Picture Coding Symp.*, vol. 62, pp. 291–294, January 2004.

[7] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *Image Processing, IEEE Transactions on*, vol. 14, no. 4, pp. 423–438, April 2005.

[8] B. Zeng and J. Fu, "Directional discrete cosine transforms for image coding," *Multimedia and Expo, 2006 IEEE International Conference on*, pp. 721–724, 9-12 July 2006.

[9] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P. Dragotti, "Directionlets: anisotropic multidirectional representation with separable filtering," *Image Processing, IEEE Transactions on*, vol. 15, no. 7, pp. 1916–1933, July 2006.

[10] C.-L. Chang and B. Girod, "Direction-adaptive discrete wavelet transform for image compression," *Image Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 1289–1302, May 2007.

[11] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transfom," *Computers, IEEE Transactions on*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.

[12] F. Kamisli and J. Lim, "Directional wavelet transforms for prediction residuals in video coding," *Image Processing, 2009. ICIP 2009. 16th IEEE International Conference on*, November 2009.

[13] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h.264/avc video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 560–576, July 2003.

[14] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *VCEG Contribution VCEG-M33*, April 2001.

[15] P. Kauff and K. Schuur, "Shape-adaptive dct with block-based dc separation and dc correction," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 8, no. 3, pp. 237–242, Jun 1998.

Fatih Kamisli Fatih Kamisli is a PhD candidate in the Advanced Telecommunications and Signal Processing group at MIT. He received the B.S. degree in electrical and electronics engineering from the Middle East Technical University in 2003, and the S.M. degree in electrical engineering and computer science from the Massachusetts Institute of Technology in 2006. His current research interests include video processing/compression and digital signal processing.

Jae S. Lim Jae S. Lim received the S.B., S.M., E.E., and Sc.D. degrees in EECS from the Massachusetts Institute of Technology in 1974, 1975, 1978, and 1978, respectively. He joined the M.I.T. faculty in 1978 and is currently a Professor in the EECS Department.

His research interests include digital signal processing and its applications to image and speech processing. He has contributed more than one hundred articles to journals and conference proceedings. He is a holder of more than 30 patents in the areas of advanced television systems and signal compression. He is the author of a textbook, Two-Dimensional Signal and Image Processing. He is the recipient of many awards including the Senior Award from the IEEE ASSP Society and the Harold E. Edgerton Faculty Achievement Award from MIT. He is a member of the Academy of Digital Television Pioneers and a fellow of the IEEE.