12

Spatial prediction in the H.264/AVC FRExt coder and its optimization

Simone Milani
Dept. of Information Engineering, University of Padova
Italy

1. Introduction

Intra-only video coding is a widely used coding method in professional and surveillance video applications. This fact is partly due to its ease of editing and partly due to the significant amount of computational complexity required by motion estimation, which results hard to adopt in real-time video systems. In the H.264/AVC standardization process (Richardson, 2003) the compression performance of Intra coding was significantly improved by the adoption of spatial prediction in Intra frames, which have permitted the H.264/AVC coder to obtain a higher compression gain with respect to the previous coding standards, like JPEG2000 (Cho et al., 2007). The pixels of the current block are predicted using the reconstructed pixels of neighboring blocks interpolated along different orientations, which result closely related to the characteristics of the image correlation (Cappellari and Mian, 2004). In the first version of the H.264/AVC standard (Joint Video Team, 2002), the spatial prediction is limited to either blocks of 4×4 pixels or whole macroblocks (MBs) of 16×16 pixels. In the FRExt extension of the standard (Joint Video Team, 2004), blocks of 8×8 pixels are considered too. As a consequence, the computational complexity of an exhaustive rate-distortion optimization significantly increases because of the number of different partitioning modes and prediction directions. In order to overcome this problem, a wide variety of complexity reduction strategies, together with the introduction of novel hardware accelerators, have been proposed in literature.

Pan et al. (2005) propose a fast Intra prediction algorithm that extracts the image features using Sobel edge operators and chooses the predictor according to their statistics. In a similar way, the approaches by Pan et al. (2004) and by Ryu and Kim (2007) extract the directional features of each frame and use them to estimate the most probable prediction modes. The solutions proposed by Xin et al. (2004) and by Jeong and Kwon (2007) evaluate the distortion produced by prediction in the transform domain, while (Kim et al., 2006a) suggest extracting jointly the features of each block from both pixels and transform coefficients. In addition, temporal correlation existing between adjacent frames can be used too, as it is shown by Xin and Vetro (2006).

Many approaches employ early-termination decision in order to reduce the amount of computation (Lorás and Amiel, 2005). This makes the computational complexity significantly vary according to the processed video sequence (see the strategy proposed by Yong-dong et al. (2004) as an example where the relative reduction of coding time varies from 40% to 70%), and therefore, an *a priori* estimation of the resulting cost is not possible. At the same time, the performance of the algorithm varies according to the coded sequence like in the case of the approach by Kalva and Christodoulou (2007) where a machine learning algorithm is used to select the best prediction mode among a reduced set of candidates.

With respect to these methods, the design of a complexity reduction strategy that permits controlling the amount of required computation provides several advantages, such as

- the possibility of adapting the algorithm to devices with different computational capabilities and power supply;
- an accurate estimation of the autonomy of battery-powered coding devices;
- the possibility of enabling power saving configurations that gradually reduce the computational complexity (at the cost of a worse rate-distortion optimization) according to the remaining battery charge.

The solution presented in the following computes for each 4×4 prediction mode the probability that it minimizes the cost function with respect to the other ones. According to this probability distribution, the algorithm elects a limited set of modes (the most probable ones) as possible *"best-mode"* candidates and computes the cost function for each of them. The probability estimation is performed using a low-cost Belief-Propagation (BP) strategy that exploits the statistical dependence among adjacent blocks. In the end, the algorithm checks whether it is worth merging the blocks together or not.

In the following, Section 2 will describe the Intra prediction process in the H.264/AVC FRExt coder and the related complexity problems. Section 3 presents how different solutions try to cope with the computational issue by designing appropriate low-complexity Intra prediction strategies. Then, Section 4 will present the proposed algorithm that estimates the best-mode probability for each prediction orientation from the previous coding results. The estimated probability distribution permits computing a reduced set of candidate modes. In second step, the algorithm chooses whether it is worth merging blocks together or not. Experimental results, presented in Section 5, will show that the performance of the algorithm compares well with other solutions, and in addition, that the computational complexity can be controlled by increasing or decreasing the number of candidate modes. Final conclusions will be drawn in Section 6.

2. The Intra coding mode in the H.264/AVC FRExt standard

Like many of the previously-proposed video coders (ISO/IEC JTC1, 2001; ITU-T, 1995; ITU-T and ISO/IEC JTC1, 1994), the H.264/AVC standard adopts a hybrid coding scheme that combines traditional transform coding with a predictive coding approach (see Fig. 1). The adoption of low-complexity integer transform decreases the compression efficiency of the transformation procedure for the sake of a lower computational complexity; it is possible to compensate this decrement in the coding gain by predicting the input signal from pixels belonging to the previous frames or to the previously-coded blocks.

The input video frames captured by the camera are partitioned into macroblocks of 16×16 pixels, and each macroblock can be divided into blocks of 4×4 or 8×8 pixels which can be predicted according to the chosen coding mode. For Inter macroblocks (i.e. temporally predicted macroblocks), from the previously-coded frames the coder selects a predictor block that approximates well the current one and identifies it using a motion vector (MV). This selection is performed via a motion search process, which considers all the blocks whose coordinates lie within a given search window and chooses the one that minimizes a given distortion metric. As for Intra macroblocks, the current block is predicted using the neighboring pixels

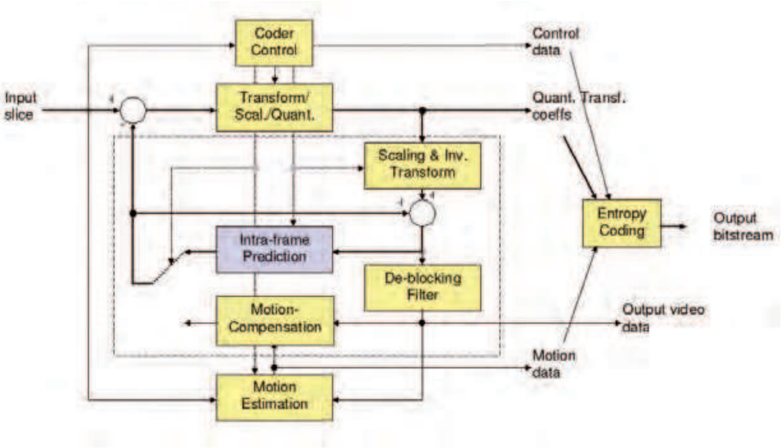


Fig. 1. Block diagram of the H.264/AVC coder.

belonging to previously-coded blocks and interpolating their values according to a set of linear equations. The residual signal after prediction is then transformed and quantized into a set of integer coefficients whose values are converted into a binary bit stream. The size of the adopted integer transform can be either 4×4 or 8×8 depending on the chosen macroblock partitioning. Since the adopted transforms are not orthonormal, the quantization unit needs to compensate this fact by rescaling the quantization steps for the different coefficients depending on the spatial frequencies. As a consequence, the set of quantization steps associated to a given distortion level is referenced using the Quantization Parameter QP, which assumes integer values in the range $[0\ 51]$ and is exponentially proportional to the quantization step according to the equation

$$\Delta = K_{i,j}\, 2^{QP/6} \tag{1}$$

where $K_{i,j}$ is a rescaling factor that depends on the spatial frequencies (i, j) and on the adopted quantization matrix. Then the block of coefficients is dequantized, inversely- transformed, and summed to the corresponding predictor block in order to reconstruct the coded signal. In the following, the chapter will be focused on the spatial Intra prediction.

Since the earliest stages of its standardization process, the Intra coding mode of the H.264/AVC codec has been characterized by block-based spatial prediction. The pixels in the current block are predicted from the neighboring ones according to a spatial predictor which is chosen among a set of possible standardized candidates (see Fig. 2 as an example for the `Intra4x4` mode).

More precisely, each candidate predictor is computed from the neighboring pixels of the upper and the left macroblocks interpolated along an assigned spatial directions. The H.264/AVC standard defines a finite set of directions whose number can vary from 4 up to 9 according to the coding mode of the block. At first, two Intra coding modes were defined, named `Intra4x4` and `Intra16x16` respectively. The first one performs spatial prediction on blocks of 4×4 pixels and has a set of 9 candidate predictors (reported in Fig. 2), while the second one predicts a whole macroblock of 16×16 pixels choosing one predictor among a set of 4 (see Fig. 3). As for the Chroma component, only 4 modes were standardized defining an Intra prediction on 8×8 blocks. With the extension of the coding standard (H.264/AVC FRExt), a novel `Intra8x8` mode was introduced using 9 possible candidates on Luma blocks of 8×8 pixels (Joint Video Team, 2004). Experimental results (Cappellari and Mian, 2004) have shown that the performance of spatial prediction coding in the H.264/AVC coder depends on the efficiency of the chosen directional predictor in modelling the characteristics of the signal. Given the Intra macroblock coding mode M ($M = \texttt{Intra4x4}, \texttt{Intra8x8}, \texttt{Intra16x16}$), the default Intra coding strategy implemented in the reference H.264/AVC coder tests for the current block all the possible predictor blocks associated to the set of available prediction modes and chooses the mode m that minimizes the Lagrangian cost function

$$L(m) = D(m) + \lambda\, R(m). \tag{2}$$

The value $R(m)$ is the bit rate needed to code the current mode m in the bit stream, $D(m)$ is the distortion metric, and λ is a Lagrange multiplier that weights the influence of both distortion and bit rate in the cost function (Sullivan and Wiegand, 1998). In finding the best prediction mode, the distortion $D(m)$ is measured using the Sum of Absolute Differences (SAD) between the predicted block and the original one in order to limit the required computational complexity. Since the best prediction mode m for the current 4×4 block is strongly correlated with the modes chosen for the spatially-neighboring blocks, in the H.264/AVC standard the bit rate $R(m)$ is coded after estimating the most probable prediction mode according to the modes of the upper and left blocks (Joint Video Team, 2004). The same rate distortion strategy is adopted to find the best Intra macroblock coding mode M, where the distortion metric $D(M)$ is the Sum of the Squared Differences (SSD) between the original macroblock and the reconstructed one in place of the SAD. Depending on the adopted distortion metric (SAD or SSD), the value of the parameter λ is linearly or quadratically proportional to the adopted quantization step. The derivation process of the parameter λ for the reference H.264/AVC coder is reported in a work by Wiegand and Girod (2001), where λ is set to

$$\lambda = 0.85\ 2^{QP/6} \text{ when using SAD and } \lambda = 0.85\ 2^{QP/3} \text{ when using SSD.} \tag{3}$$

The distortion metric in eq (2) permits choosing the coding mode that could be slightly suboptimal in terms of distortion but requires a lower amount of bits.
In the following, the chapter will give a more detailed description about each prediction mode, how it is chosen, and how it is coded in the bit stream.

2.1 Spatial prediction on 4×4 blocks

The baseline Intra coding mode defined within the H.264/AVC standard is the `Intra4x4` mode, which partitions the macroblock into 4×4-pixels blocks and chooses for each block a spatial predictor out from a set of 9 possible standardized candidates. In Figure 2 all the 9 possible modes are reported (for a detailed description including the formulas to estimate the pixel values of each predictor see the standard release by Joint Video Team 2004). Note that each mode is associated to an identification number that is closely related to its average best-mode probability, i.e. the probability of being chosen as best prediction mode for the current block. As a consequence, vertical and horizontal modes, which are the most frequently adopted prediction orientations, are assigned to the numbers 0 and 1. Since the best prediction mode m for the current 4×4 block is strongly correlated with the modes chosen for the spatially-neighboring blocks, in the H.264/AVC standard the bit rate $R(m)$ is coded after estimating the most probable prediction mode according to the modes of the upper and left blocks (see Joint Video Team (2004)). The variable `most_probable_mode` is defined as the minimum between the modes of the neighboring upper and the left blocks. Whenever estimating

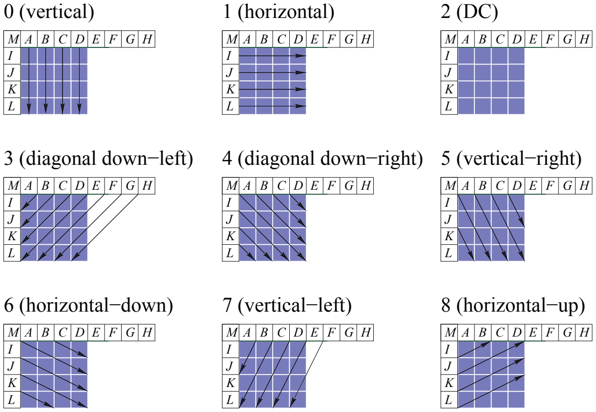


Fig. 2. Possible predictors for `Intra4x4` macroblock coding mode of H.264/AVC standard.

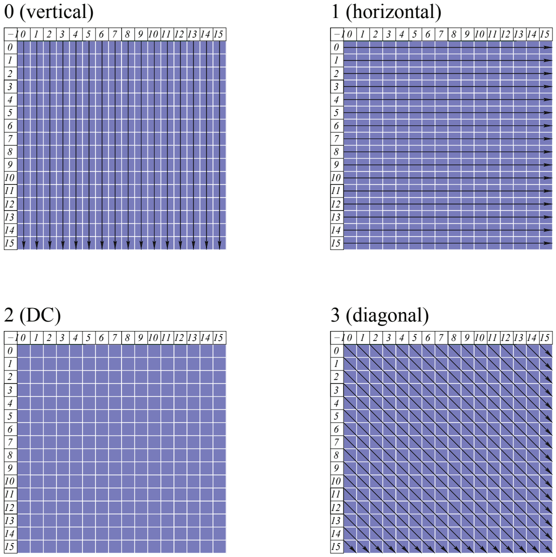
the best prediction mode m for the current block, the parameter $R(m)$ is equal to 1 or 4 since the coder at first signals with one bit whether the chosen mode equals `most_probable_mode` or not. In case m and `most_probable_mode` differs, the coder requires 3 additional bits to signal the correct predictors.

2.2 Spatial prediction on 16×16 blocks

The adoption of spatial predictions on 4×4 blocks imply that the video coder has to specify 16 prediction modes for each `Intra4x4` macroblocks with a consequent waste of coded bits. In addition, during the standardization process of the H.264/AVC architecture, preliminary experimental results had shown that spatial orientation of neighboring blocks does not change wherever the image is highly stationary (uniform regions). In these situations, the same results had also shown that performing spatial prediction on wider blocks proved to be more effective since a reduced number of prediction modes need to be specified. As a consequence, the H.264/AVC video coding standard was enabled with the additional `Intra16x16` coding mode, which predicts the whole macroblock using the pixels of the upper and the left macroblocks lying along the borders. In this case, 4 possible orientations were defined and, after performing the 4×4 integer transform on each residual block within the macroblock, an additional 4×4 Hadamard transform is applied on the DC coefficients (Joint Video Team, 2004). Figure 3 reports a graphic representation of the possible Intra prediction modes. Anyway, the extension of the H.264/AVC standard to other applications and video formats brought the need of defining an additional Intra prediction mode operating on the intermediate 8×8 blocks.

2.3 Spatial prediction on 8×8 blocks

Initially conceived for video communication and video streaming applications on low bandwidth channels, the standard H.264/AVC was successively extended to the transmission and storage of high definition video. As a drawback, the 4×4 integer transform was no more suitable for wider video formats, and therefore, an additional 8×8 integer transform was included in the standard. In addition, experimental results showed that performing the block-based spatial prediction on 8×8 blocks proved to be effective for a wide number of mac-



0 (vertical)

1 (horizontal)

2 (DC)

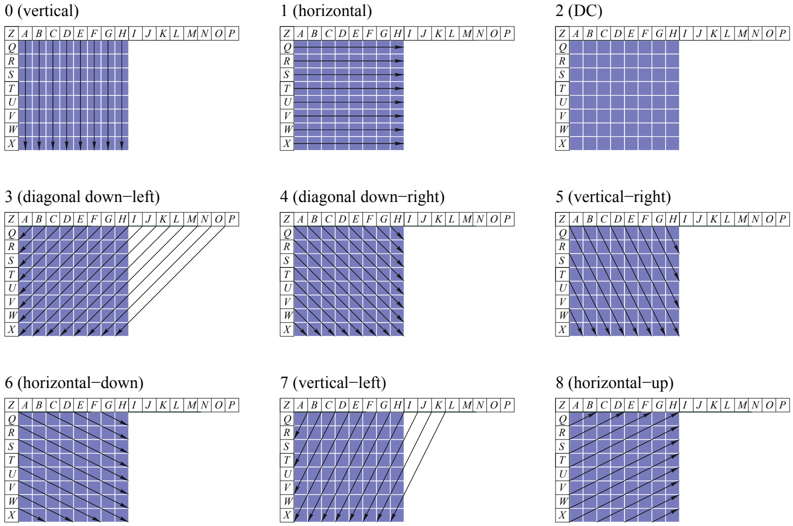
3 (diagonal)

Fig. 3. Possible predictors for `Intra16x16` macroblock coding mode of H.264/AVC standard.

roblocks. As a result, the `Intra8x8` coding mode was introduced within the standard as an additional feature which has to be included in all the decoders implementing the highest profiles (Joint Video Team, 2004). The Intra prediction on 8 × 8 blocks has 9 different prediction orientations similar to those defined for the 4 × 4 blocks. In this case, the neighboring pixels are initially smoothed by a low-pass Finite Impulse Response (FIR) filter and then used to generate the predictor blocks. The coding strategy for the prediction mode is the same for the mode `Intra4x4` (see Fig. 4). In this case, the variable `most_probable_mode` can be computed from neighboring 4 × 4 blocks too (see the standard draft by Joint Video Team, 2004 for more details).

2.4 Spatial prediction on chrominance blocks

For input video signals in the YUV 4:2:0 format, which is the main format supported by the standard H.264/AVC (however, other formats are supported), every macroblock of luminance pixel is associated to two 8 × 8 pixel blocks in the U and V components respectively. Spatial prediction is also performed on these blocks considering neighboring pixels from the upper and the left macroblock. In this case, the characteristics of the chrominance signals do not require a wide range of possible predictor modes to perform an effective spatial prediction, and therefore, only 4 modes are defined within the standard similar to those defined for the `Intra16x16` mode despite the fact that the indexing is changed (see Fig. 3).
The default Intra coding algorithm tests all these possible choice and chooses the one that provides the best rate-distortion performance in terms of the metric of eq. (2). In the following we will present some solutions that permit obtaining a coding efficiency comparable to that of the extensive method and require a limited computational complexity.



0 (vertical) 1 (horizontal) 2 (DC)

3 (diagonal down−left) 4 (diagonal down−right) 5 (vertical−right)

6 (horizontal−down) 7 (vertical−left) 8 (horizontal−up)

Fig. 4. Possible predictors for `Intra8x8` macroblock coding mode of H.264/AVC FRExt standard.

3. Overview of the existing Fast Intra Prediction methods

Most of the fast Intra coding algorithms reduce the computational complexity by identifying the spatial orientation of the current block and selecting an appropriate set of candidate modes without performing a complete testing of all the possible predictors (Pan et al., 2005). At the same time, these algorithms select the MB partitioning (`Intra4x4` or `Intra16x16`) that suits better to the current macroblock. Many methods rely on early termination solution, where the best predictor search is terminated before testing all the available modes whenever the resulting distortion is lower than a given threshold. Other solutions analyze the characteristics of the block to be coded in order to identify the most suitable prediction mode. These methods test the different predictors according to a hierarchical order or process the input block using some edge detection operators to create a set of possible candidate modes. The outcomes of these operations are used to infer the spatial orientations of the block to be coded and identify, as a consequence, the associated prediction mode. In most of the solutions presented in literature, the number of tested modes or operations depend on the characteristics of the input video signal, and as a consequence, the required computational complexity varies without permitting an *a priori* estimation. In the following some of these methods are shortly presented.

3.1 Fast Intra Prediction using edge detection operators

Since the efficiency of the spatial prediction relies on identifying accurately the orientation of the spatial correlation, many approaches try to infer this feature from the coded video signal using some edge detection operators. One of the first approaches that perform fast Intra prediction estimation was proposed by Pan et al. (2005). When evaluating the `Intra4x4` mode, the input 4×4 block is processed using vertical and horizontal Sobel operators, and the outcoming values are stored in a histogram with 9 bins associated to the different modes.

These values are used to estimate a set $\mathcal{M}$ of possible candidate modes. The mode with the highest probability, together with the two modes that result the closest to it and the DC prediction mode, is included in $\mathcal{M}$. Then, the algorithm chooses the predictor that obtains the lowest value in the cost function. As for the `Intra16x16` mode, only the most probable mode and the DC mode are considered reducing the number of tested modes from 4 to 2. When coding the 8×8 chrominance blocks, the most probable modes are estimated for both the U and V components from the orientation histograms and included in $\mathcal{M}$, together with the DC mode.

A similar approach is adopted in another work by the same authors (Pan et al., 2004), where an average edge directional field is computed for each 4×4 block in order to identify the dominant spatial orientation. The quadratic values of the Sobel operators are averaged within the current block in order to estimate the dominant spatial direction and the related coherence value.

A different approach is proposed by Yong-dong et al. (2004), which generates a subsampled version of the current 4×4 block and computes vertical and horizontal edge detection operators. According to the absolute values and the signs of these, different sets of candidate predictors (with different numbers of modes included) are generated and tested. In this way it is possible to save about 60 % of the computational complexity on average, but the actual saving depends on the chosen quantization parameter QP.

Other approaches approximate the distortion measure using alternative metrics, which either requires a lower complexity or proves to be more effective in identifying the orientation of the current block. The approach proposed by Kim et al. (2006b) evaluates a group of SAD metrics on a reduced set of pixels which are located close to the borders of each block. A possible alternative is presented by Jeong and Kwon (2007) where the orientation of the block is found computing a distortion metric on the relevant transform coefficients of the current block.

3.2 Fast Intra Prediction using hierarchical search

Another set of solution proposed in literature rely on the possibility that a tested prediction mode is very likely to be the best one whenever the associated distortion value is lower than a discriminating threshold. As a consequence, these solutions aim at finding the mode order that places the most probable best candidates first.

The approach by Lu and Yin (2005) tests the available prediction modes and coding options according to a predefined order. More precisely, the `Intra16x16` mode is considered at first, and the coding strategy tests the DC mode checking whether the associated cost function has a lower value with respect to fixed discriminating threshold. In case the cost is higher, vertical and horizontal modes are tested as well; otherwise, the `Intra16x16` coding process is finished (*early termination*). The `Intra4x4` is then tested considering DC, vertical, and horizontal modes at the beginning. In case the cost function for the `Intra16x16` mode is lower than a given threshold no additional `Intra4x4` prediction modes are considered.[1] Otherwise, the algorithm tests the remaining prediction orientations that are closer to the best mode between the vertical and the horizontal ones. The presence of early termination decision does not allow an accurate *a priori* estimation of the required computational cost.

In a similar way, the solution designed by Lorás and Amiel (2005) tests the vertical and the horizontal directions first, and according to whether the vertical or the horizontal orientation is better, it chooses the following set of modes to test. The same policy is applied to the new

---

[1] Early termination for the DC mode is evaluated also in this case.

candidate modes following a tree-ordered refinement policy of the Intra prediction for the current block.

Another hierarchical solution was proposed by Kalva and Christodoulou (2007), where the modes are tested following an adaptive tree structure that is modified using a machine learning algorithm.

3.3 Fast Intra Prediction using parametric models

Among the strategies that reduce the required computational complexity, a separate mention has to be done for those strategies that aim at achieving a lower computational cost the coding performance of the rate-distortion optimization algorithm proposed within the standard (Wiegand and Girod, 2001). The rate and the distortion of the final coded block are estimated using some parametric models. This class of algorithms compute some low-complexity metrics that characterize the features of the original signal, and use them to estimate the final coded bit rate and the associated distortion, whose calculation requires a significant amount of operations.

The approach proposed by Kim et al. (2006a) estimates the possible results of the rate-distortion optimization algorithm from the SAD metric computed on the pixel blocks and on the blocks obtained after a Hadamard transform (in this case the SAD is called SATD). The SAD and the SATD values permit identifying the prediction mode that is the most likely to be the best one. In a similar way, the strategy by Kim et al. (2003) infers a statistical model for the current block from the SATD values.

Unfortunately, many of these solutions adopt early termination strategies that make the required computational complexity vary. In the following we will present an optimization approach that permits controlling the amount of calculation with deterministic accuracy. In this way, it is possible to configure the algorithm in a flexible way according to the desired computational complexity.

4. A low-complexity Belief Propagation based Intra prediction strategy

The approach proposed by Milani (2008) reduces the set of tested candidate modes according to a probability estimation strategy, which is based on a Belief Propagation algorithm. This solution can be divided into three parts. At first, the algorithm estimates the most probable orientations for the current block. The estimated probabilities are used to generate a set of candidate predictors, and the best prediction mode is found by coding the current MB using the `Intra4x4` mode. In the following, the 4×4 blocks are fused into either 8×8 blocks or a whole 16×16-pixels macroblock according to their orientations. The following sections will present the three phases in detail.

4.1 Probability estimation for the best candidate modes

4.1.1 Estimation of orientations for 4×4 blocks

Assuming that the $M_0 \times 1$ array $\mathbf{p}(x,y) = [p_m(x,y)]$ $(m = 0, \ldots, M_0 - 1)$ groups the probabilities $p_m(x,y)$ that the mode m is the best mode for the block at coordinates (x,y) (with M_0 the total number of candidate modes), it is possible to write the elements of $\mathbf{p}(x,y)$ as follows

$$p_m(x,y) = \mathbf{p}^T(x,y-1)\, Q^m(x,y)\, \mathbf{p}(x-1,y), \tag{4}$$

where $Q^m(x,y) = [q^m_{i,j}(x,y)]$ is an $M_0 \times M_0$ matrix. The value $q^m_{i,j}(x,y)$ represents the conditional probability that mode m is the best mode for the current block at (x,y) given that i and

j are the best modes for blocks at coordinates $(x, y-1)$ and $(x-1, y)$ respectively. However, it may happen that only a smaller set $\mathcal{M}$ of M candidate modes ($M < M_0$) are available for the block at (x, y), and therefore, the probabilities $p_{m'}(x, y)$ are 0 for $m' \notin \mathcal{M}$. This is the case of blocks placed at positions where some reference pixels are not available because of the frame boundaries or the block coding order (e.g. upper-right pixels can not be used since the corresponding neighboring block has not been coded yet). The same candidate modes reduction is found for all the blocks whenever the H.264/AVC coder adopts a fast intra prediction algorithm that tests only a selected set of candidates to constrain the computational complexity. This candidate modes reduction affects the best-mode probability array, which can be replaced with the relation

$$\tilde{\mathbf{p}}(x, y) = P_M(x-1, y)\ \mathbf{p}(x, y) \tag{5}$$

where $P_M(x, y)$ is a singular projection matrix that sets to 0 some elements of $\mathbf{p}(x, y)$ according to which candidate modes are available.

As a consequence, the best-mode statistics for the current block at position (x, y) can be estimated propagating the best-mode probability of previous blocks via the equation

$$\begin{aligned} \tilde{p}_m(x, y) &= \mathbf{p}^T(x, y-1)\ P_M^T(x, y-1)\ Q^m(x, y)\ P_M(x-1, y)\ \mathbf{p}(x-1, y) \\ &= \tilde{\mathbf{p}}^T(x, y-1)\ Q^m(x, y)\ \tilde{\mathbf{p}}(x-1, y) \end{aligned} \tag{6}$$

(which is a modified version of eq. (4)), and projecting the array $\mathbf{p}(x, y)$ onto the subspace of allowed modes using equation (5). The resulting array $\tilde{\mathbf{p}}(x, y)$ differs from the original estimate $\mathbf{p}(x, y)$ of eq. (4) because of the approximation introduced by the projection and leads to a different set $\tilde{\mathcal{M}} \neq \mathcal{M}$ of candidate modes. As a possible drawback, the chosen predictor could not match accurately the orientation of the local correlation either because the optimal mode is not included in the set $\tilde{\mathcal{M}}$ or because all the required neighboring pixels are not available and the most appropriate predictor can not be adopted. The finally chosen mode $\tilde{m}$ could result sub-optimal for the current block and is going to affect the accuracy of probability estimation for the following adjacent blocks. It is possible to mitigate this effect by adopting a Belief-Propagation (BP) strategy that refines the statistics for each block.

4.1.2 The Belief-Propagation procedure for spatial orientations of 4×4 blocks

Before coding the block at the coordinates (x, y), the mode estimation routine propagates through a BP procedure the information about the best modes for the upper and left blocks found during the coding operations (see Figure 5 for a graphic example). These modes are denoted here with $\tilde{m}(x, y-1)$ and $\tilde{m}(x-1, y)$ respectively. According to this, the coding routine estimates a probability mass function (pmf) $\tilde{\mathbf{p}}(x, y)$ for the current block via equation (6), where

$$\begin{aligned} \tilde{p}_m(x-1, y) &= \begin{cases} 0 & m \neq \tilde{m}(x-1, y) \\ 1 & m = \tilde{m}(x-1, y) \end{cases} \\ \tilde{p}_m(x, y-1) &= \begin{cases} 0 & m \neq \tilde{m}(x, y-1) \\ 1 & m = \tilde{m}(x, y-1). \end{cases} \end{aligned} \tag{7}$$

According to the values of $\tilde{\mathbf{p}}(x, y)$, all the possible prediction modes are sorted in decreasing probability order, and the most probable ones are included in the set $\mathcal{M}$ according to the criteria that will be described in Section 4.2.

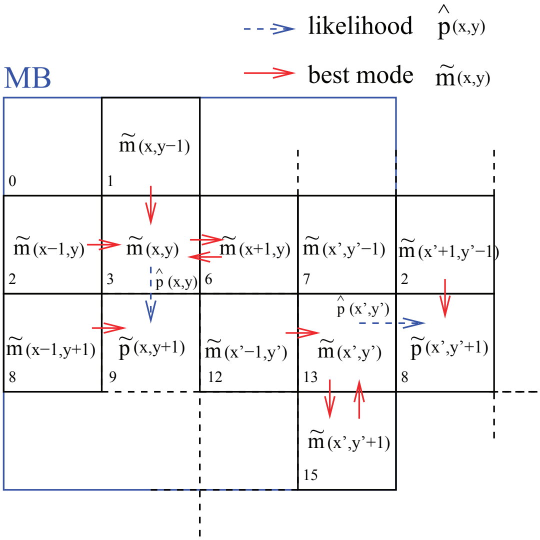


Fig. 5. Probability propagation according to the implemented Belief Propagation approach. Some message passing propagates hard information (solid arrows) regarding the chosen prediction modes, while others communicates likelihoods associated to the prediction mode of 4 × 4 blocks (dashed arrows).

After finding the mode that minimizes the cost function among the candidates in $\mathcal{M}$, the BP approach propagates this result to the previously coded blocks in order to refine the accuracy of the estimated mode probability (i.e. $\tilde{\mathbf{p}}(x, y-1)$ and $\tilde{\mathbf{p}}(x-1, y)$). The array $\tilde{\mathbf{p}}(x, y)$, whose elements are reported in eq. (7), is replaced by a *"soft"* best mode estimation $\hat{\mathbf{p}}(x, y)$ (a likelihood) computed using a reversed version of equation (4)

$$\hat{\mathbf{p}}(x, y) = \tilde{\mathbf{p}}^T(x, y-1)\; Q^{m,r}(x, y)\; \tilde{\mathbf{p}}(x+1, y). \tag{8}$$

The new arrays $\hat{\mathbf{p}}(x, y)$ affect the estimated mode probability distribution for the following blocks and improve the compression performance of the fast Intra coding algorithm. As an example, the elements for the probability array $\tilde{\mathbf{p}}(x, y+1)$ of block 9 in Fig. 5 are obtained via eq. (6) replacing the array $\tilde{\mathbf{p}}(x, y-1)$ with $\hat{\mathbf{p}}(x, y-1)$.
Experimental results have proved that the refinement step performed using equation (8) does not change the arrays $\tilde{\mathbf{p}}(x, y)$ in such a way that the order of candidate modes is altered. However, the likelihood estimate for the prediction mode $\tilde{m}(x, y)$ proves to be significant in the computation of the number of candidate modes as it will be explained in Subsection 4.2.1. Moreover, the best prediction modes found for `Intra4x4` coding are used to characterize the best-mode probability of prediction modes for bigger blocks in case the rate-distortion algorithm has chosen to merge the 4 × 4 blocks together, as it will be described in Section 4.4. In the estimation routine, the arrays $\tilde{\mathbf{p}}(x, y)$ and $\hat{\mathbf{p}}(x, y)$ are approximated using a finite set $\mathcal{P}$ of 100 pmfs, which has been obtained from an extensive set of training sequences via an LBG iterative classification (Gersho and Gray, 1991). In this procedure the distortion metric to

minimize is the Jensen-Shannon divergence between $\tilde{\mathbf{p}}(x,y)$ and $\hat{\mathbf{p}} \in \mathcal{P}$

$$JSD\left(\tilde{\mathbf{p}}(x,y)\|\hat{\mathbf{p}}\right) = \frac{1}{2} D\left(\tilde{\mathbf{p}}(x,y)\|\hat{\mathbf{p}}\right) + \frac{1}{2} D\left(\hat{\mathbf{p}}\|\tilde{\mathbf{p}}(x,y)\right) \tag{9}$$

where $D\left(\tilde{\mathbf{p}}(x,y)\|\hat{\mathbf{p}}\right)$ is the Kullback-Leibler divergence

$$D\left(\tilde{\mathbf{p}}(x,y)\|\hat{\mathbf{p}}\right) = \sum_{m=0}^{8} \tilde{\mathbf{p}}_m(x,y) \log\left(\frac{\tilde{\mathbf{p}}_m(x,y)}{\hat{\mathbf{p}}_m}\right). \tag{10}$$

The conditional probability matrix $Q_m(x,y)$ is a linear combination of arrays $\hat{\mathbf{p}} \in \mathcal{P}$, and the probability array $\tilde{\mathbf{p}}(x,y)$ at position (x,y) is updated after each iteration of the BP procedure using a Finite State Machine (FSM), where each state is related to an element of $\mathcal{P}$. In this way, it is possible to obtain an adaptive estimation of the probability for each prediction mode with limited computational complexity and memory area.

4.1.3 Estimation of probable spatial orientations for 8×8 blocks

After the optimization algorithm has chosen to merge together 4×4 blocks into 8×8 blocks, the coder estimates an `Intra8x8` best-mode probability distribution $\mathbf{p}^{8\times 8}$ according to the previously found `Intra4x4` modes. The adopted approach estimates three different mode probability distribution $\mathbf{p}^{8\times 8,i}$, $i = v, h, d$, which are dependent on the best Intra prediction modes of vertical, horizontal and diagonal couples of 4×4 blocks respectively (see Figure 6). Using the same notation of equation (4), it is possible to write $\mathbf{p}^{8\times 8,i} = [p_m^{8\times 8,i}]$, $i = v, h, d$ and $m = 0, \ldots, 8$, as

$$\begin{aligned} p_m^{8\times 8,v} &= \tilde{\mathbf{p}}^T(x,y)\; F_m^{8\times 8,v}\; \tilde{\mathbf{p}}(x,y+1) \\ p_m^{8\times 8,h} &= \tilde{\mathbf{p}}^T(x,y)\; F_m^{8\times 8,h}\; \tilde{\mathbf{p}}(x+1,y) \\ p_m^{8\times 8,d} &= \tilde{\mathbf{p}}^T(x,y)\; F_m^{8\times 8,d}\; \tilde{\mathbf{p}}(x+1,y+1) \end{aligned} \tag{11}$$

where $\tilde{\mathbf{p}}(x,y)$ represents the chosen prediction mode (as defined in equation (7)) and $F_m^{8\times 8,i}$, $i = v, h, d$, is the conditional probability matrix of 8×8 Intra prediction mode m given the vertical, horizontal, and diagonal couples of 4×4 modes. In this way, `Intra8x8` best-mode probability estimation relies on the results of `Intra4x4` coding which has already been performed on the current macroblock.

4.2 Estimation of the set of candidates

4.2.1 Computation of the most probable prediction modes

After estimating the probability array $\tilde{\mathbf{p}}(x,y)$, the coding routine has to identify those modes that are more likely to be the best prediction mode for the current 4×4 block. The number of candidate modes M is usually set to the average value $\overline{M}$, but can vary according to the characteristics of the probability distribution identified by $\tilde{\mathbf{p}}(x,y)$. In fact, experimental data show that the entropies of distributions $\tilde{\mathbf{p}}(x,y)$ vary, and therefore, the mode probability distributions with a lower entropy only needs a reduced number of candidates.

Named $\mathbf{ß} = [\mathbf{ß_m}]$ the average mode probability array, M is chosen in such a way that

$$\begin{aligned} \sum_{m=0}^{M-1} S_m(\tilde{\mathbf{p}}(x,y)) \; &\le \; \sum_{m=0}^{\overline{M}-1} S_m\left(\mathbf{ß}(\mathbf{x},\mathbf{y})\right) \\ \sum_{m=0}^{M} S_m(\tilde{\mathbf{p}}(x,y)) \; &> \; \sum_{m=0}^{\overline{M}-1} S_m\left(\mathbf{ß}(\mathbf{x},\mathbf{y})\right) \end{aligned} \qquad (12)$$

where $S_m(\cdot) : [0,1]^9 \;\; \rightarrow [0,1]$ is an ordering function that returns the m-th value of the input array in decreasing order. The value $\overline{M}$ reports the average number of modes to be tested for each block and permits controlling the computational complexity. In this way it is possible to provide the same probability of finding the best prediction mode with a limited sets of candidates to all the 4×4 blocks of the image. This equalization permits saving some computational complexity without affecting the coding performance of the algorithm.

As it was mentioned in Subsection 4.1.2, the refinement provided by either $\tilde{\mathbf{p}}(x,y+1)$ or $\tilde{\mathbf{p}}(x+1,y)$ permits a better estimate of the probabilities related to the candidate modes of the current block. This improvement does not lead to a change in the order of modes but could modify the number of candidates that is considered for the current 4×4 block since it affects equation (12). Experimental results have shown that the refinement brought by the Belief-Propagation strategy leads to a reduction of the coding time with respect to the case when forward message passing is allowed only (see the solid-line arrows in Fig. 5). The same approach is adopted to estimate the set of candidate modes for `Intra8x8` as it will be described in Subsection 4.3.

4.2.2 Further reduction of the possible candidates (DD algorithm)

According to the probability values of $\tilde{\mathbf{p}}(x,y)$, the M most probable modes are included in the set $\mathcal{M}$ of candidates. Whenever the entropy associated with $\tilde{\mathbf{p}}(x,y)$ is high, it is possible that the set $\mathcal{M}$ includes modes with orthogonal spatial orientations. This fact is mainly due to the transient period in the probability estimation process, which may require several iteration before converging to an accurate estimate of mode statistics. Therefore, a further reduction of the candidate modes can be obtained by estimating whether horizontal or vertical modes are dominant in the distribution $\tilde{\mathbf{p}}(x,y)$ and eliminating the dominated modes (Dominated Deletion - DD). The number of orientations in the set $\mathcal{M}$ which are close to the vertical one is compared with the number of candidate modes which have a spatial orientation close to the horizontal one. In case one of them prevails, the modes of the other type are deleted from the set $\mathcal{M}$.

This additional improvement proves to be quite effective whenever the image orientation statistics has changed, and the array $\tilde{\mathbf{p}}(x,y)$ estimated by the algorithm does not provide a sufficiently-accurate approximation of the real pmf yet. Therefore, the estimated set $\mathcal{M}$ could include some candidate modes which are orthogonal to the other ones since they could be probable candidates in the neighboring region. The DD elimination algorithm prevents this transient phase from reducing the effectiveness of the algorithm and speeds up the best intra-prediction estimation process.

However, this elimination procedure has to be constrained in order to avoid an excessive reduction of the candidate sets whenever the statistics of prediction orientations is not clearly biased on either vertical or horizontal directions. In order to avoid the deletion of probable candidate modes, the DD algorithm is performed only for modes greater than 4 whenever the number of dominated modes is greater than a certain threshold value T. In the setting

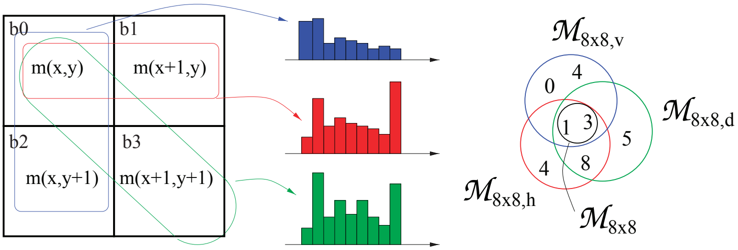


Fig. 6. Merging operation for 4 × 4 blocks.

operation of the fast intra algorithm, the parameter T can be varied in order to increase or decrease the cardinality of $\mathcal{M}$ according to the desired computational complexity.
Experimental results reported in Section 5 will underline its contribution in the overall performance of the fast intra prediction procedure.

4.3 Computation of the candidates for 8×8 blocks

In case the coding mode `Intra8x8` is enabled, the fast intra prediction algorithm has to estimate the most appropriate prediction modes for the current 8×8 block from the results of `Intra4x4` mode. The transcoding algorithm reported by Bialkowski et al. (2004) chooses the most frequently used prediction direction, but in case the estimated 4×4 orientations prove to be nonuniform within the same 8×8 block, a better performance can be obtained testing a set $\mathcal{M}_{8\times8}$ of different candidates. A procedure similar to that of Subsection 4.2.1 is adopted in order to estimate the sets of candidate $\mathcal{M}_{8\times8}$ for the current 8×8 block from $\mathbf{p}^{8\times8,i}$, $i = v, h, d$. Each mode probability distribution $\mathbf{p}^{8\times8,i}$ infers a different set $\mathcal{M}_{8\times8,i}$, $i = v, h, d$, of candidate modes which is obtained in the same way of the set of M possible candidate modes for `Intra4x4` blocks. In case the set $\mathcal{M}_{8\times8}$ obtained from the intersection of the sets

$$\mathcal{M}_{8\times8} = \mathcal{M}_{8\times8,v} \cap \mathcal{M}_{8\times8,h} \cap \mathcal{M}_{8\times8,d} \tag{13}$$

is not empty, the coding algorithm merges the 4×4 blocks into a 8×8 block and tests the predictors included in the set $\mathcal{M}_{8\times8}$ looking for the one that minimizes the cost function.
As for the `Intra16x16` coding, all the four possible predictions are tested since the estimation of the best mode probability for the 16×16 block from the best `Intra4x4` modes is not trivial. The same choice is adopted to spatially predict the chrominance components U and V.

4.4 Estimation of best macroblock partitioning for Intra prediction

After finding the best mode for each 4×4 block in the current MB, the coding routine tests whether it is better to use bigger blocks. In a first step the algorithm checks whether it is possible to merge together the 4×4 blocks into blocks of 8×8 pixels. In case the orientations of each 4×4 block are the same or close, the merging of separate blocks results convenient with respect to the `Intra4x4` block partitioning since a reduced number of predictors needs to be coded in the transmitted bit stream.
In order to detect these configurations, the encoder estimates the orientation differences $d(\tilde{m}(x,y), \tilde{m}(x+i, y+j))$ $(i, j = 0, 1)$ between vertical, horizontal and diagonal couples of 4×4 blocks (see Figure 6) within the current 8×8 block. The metric $d(\tilde{m}(x,y), \tilde{m}(x',y'))$ is computed as follows

$$d(\tilde{m}(x,y), \tilde{m}(x',y')) = \left|\angle\tilde{m}(x,y) - \angle\tilde{m}(x',y')\right| \tag{14}$$

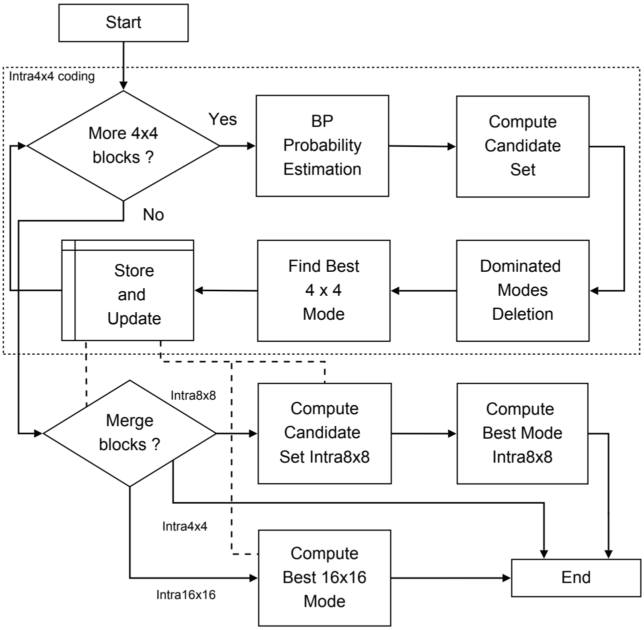


Fig. 7. Block diagram for the general Fast Intra algorithm.

where $\angle m$ denotes the angle associated to the spatial orientation of mode m. If the average difference

$$\overline{d} = \frac{d(\tilde{m}(x,y),\tilde{m}(x+1,y))}{3} + \frac{d(\tilde{m}(x,y),\tilde{m}(x,y+1))}{3} + \frac{d(\tilde{m}(x,y),\tilde{m}(x+1,y+1))}{3} \tag{15}$$

is lower than 40°, the 4×4 blocks at (x,y), $(x+1,y)$, $(x,y+1)$, and $(x+1,y+1)$ could be merged into one block of 8×8 pixels. In case the condition on $\overline{d}$ is verified for all the 8×8, the `Intra8x8` coding mode is enabled. Moreover, the encoding routine tests whether it is worth merging the 8×8 blocks into one common 16×16 prediction block considering the `Intra4x4` modes for the blocks at the border of 8×8 blocks. In case the average absolute difference between the orientations of 4×4 blocks lying at the borders of 8×8 blocks is lower than 40°, the `Intra16x16` prediction mode is chosen for the current macroblock. In this way, the wider block partitioning modes are tested only in case the orientations for the 4×4 blocks are approximately uniform, otherwise either 4×4 or 8×8 partitioning is preferred.

The whole fast Intra coding procedure is depicted in the block diagram of Fig. 7 and can be summarized by the following pseudo-code:

```
Test Intra4x4 coding mode
for each $4 \times 4$ block in the current macroblock do
    compute $\tilde{\mathbf{p}}(x,y)$
    compute $M$ and create the set $\mathcal{M}$
    test all the modes in $\mathcal{M}$ an
end for
```

```
check if it is worth merging the $4 \times 4$ blocks into bigger blocks as described in the current Section
if Intra8x8 is to be enabled then
    for each $8 \times 8$ block in the current macroblock do
        compute $\tilde{\mathbf{p}}^{8\times 8,i}$, $i = v, h, d$
        compute $\mathcal{M}_{8\times 8}$ and find the best mode
    end for
end if
if Intra16x16 mode is to be enabled then
    find the best prediction mode
end if
choose the MB Intra coding mode that minimize the total cost function.
```

Experimental results will show that this choice leads to good performance with respect to other proposed solutions.

5. Experimental results

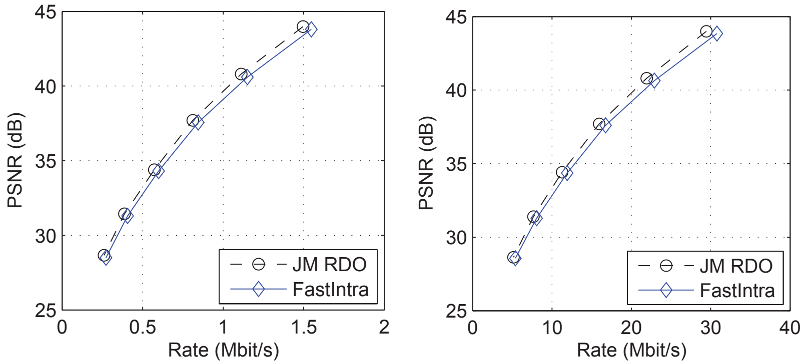
In order to test the efficiency of the presented algorithm, different sequences were coded with different quantization parameter values and enabling different Intra coding modes. The proposed Intra coding strategy was implemented into the JM10.1 software. In the tests the adopted parameter setting is the same of the paper by Pan et al. (2005), coding different sequences with only Intra frames and $QP = 28, 32, 36, 40$. At first the performance of `Intra4x4` and `Intra16x16` modes only was evaluated, comparing the computational complexity, the PSNR value, and the coded bit rate of the presented solution with those provided by the full-complexity rate-distortion optimization algorithm implemented in the reference software. Experimental data show that the PSNR vs. rate curves of the two methods are quite close (see Figure 8). It is possible to notice that coding performance in terms of rate-distortion optimization is related to the target number $\overline{M}$ of candidate modes, which can vary according to the available computational resources or the remaining power supply.

Table 1 reports the PSNR loss, together with the rate increment and the complexity reduction, for the proposed approach with respect to the reference software (with `Intra8x8` mode disabled). The presented algorithm is able to reduce the coding time of approximately 63% with respect to the JM exhaustive approach with an average rate increment lower than 5% and a PSNR loss of 0.16 dB ($\overline{M} = 6$ and $T = 2$). The DD algorithm described in Subsection 4.2.2 makes possible to improve the relative coding time reduction of an additional 13% (compare results for $\overline{M} = 6$ and $T = 2$ with results for $\overline{M} = 6$ without DD).

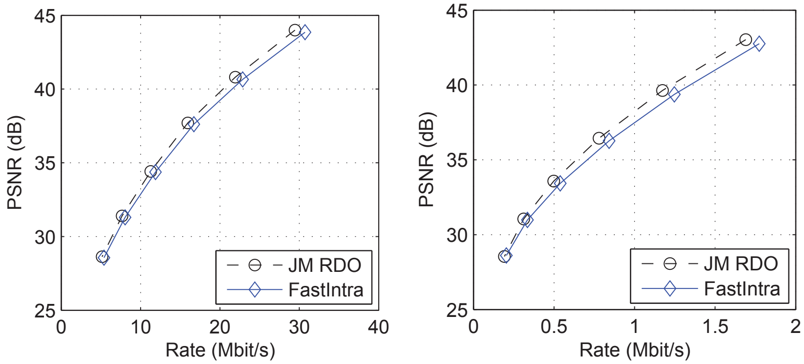
The reported data also show that the rate-distortion performance is slightly better than that of the approaches by Pan et al. (2005) and by Yong-dong et al. (2004). The bottom part of Table 1 reports the results for the algorithm proposed by Pan et al. (2005). Equalizing the rate increment, the performance of the proposed algorithm with $\overline{M} = 7$ and $T = 2$ permits reducing the PSNR loss of 0.04 dB and improving the coding time saving of approximately 2%. Despite this slight improvement, the real advantage of the proposed approach relies on the possibility of forecasting the computational complexity required by coding operations. Table 2 reports the range of variation for the saved coding time of different fast Intra coding algorithms and different configurations. It is possible to notice that the computational complexity does not significantly vary according to the input sequence, since the maximum deviation of time sav-

$(\overline{\mathbf{M}}, \mathbf{T})$	**Sequence**	**Δ Bits (%)**	**Δ PSNR**	**Δ Time (%)**
$(5, 2)$	container (qcif)	10.42	−0.18	−72.24
	news (qcif)	9.42	−0.23	−72.44
	coastguard (qcif)	8.04	−0.18	−71.55
	bus (cif)	6.22	−0.21	−70.50
	tempete (cif)	4.55	−0.30	−69.42
average		7.73	−0.22	−71.23
$(6, 2)$	container (qcif)	5.35	−0.14	−62.56
	news (qcif)	5.79	−0.16	−62.46
	coastguard (qcif)	3.55	−0.14	−62.64
	bus (cif)	2.15	−0.16	−63.64
	tempete (cif)	4.95	−0.22	−63.00
average		4.36	−0.16	−62.86
$(6, 3)$	container (qcif)	5.32	−0.14	−59.95
	news (qcif)	5.68	−0.15	−60.43
	coastguard (qcif)	3.67	−0.14	−60.48
	bus (cif)	2.15	−0.16	−63.64
	tempete (cif)	4.95	−0.22	−63.00
average		4.35	−0.16	−61.50
$(7, 2)$	container (qcif)	3.77	−0.13	−58.98
	news (qcif)	4.55	−0.16	−57.85
	coastguard (qcif)	2.14	−0.13	−59.21
	bus (cif)	2.11	−0.14	−58.19
	tempete (cif)	4.61	−0.19	−57.88
average		3.44	−0.15	−58.42
$(7, 3)$	container (qcif)	3.60	−0.12	−55.18
	news (qcif)	4.50	−0.14	−54.14
	coastguard (qcif)	2.19	−0.12	−55.25
	bus (cif)	2.13	−0.13	−54.83
	tempete (cif)	4.57	−0.18	−54.36
average		3.40	−0.14	−54.75
6 without DD	container (qcif)	5.64	−0.08	−50.43
	news (qcif)	4.96	−0.09	−47.79
	coastguard (qcif)	3.64	−0.09	−51.10
	bus (cif)	2.73	−0.10	−49.41
	tempete (cif)	5.05	−0.13	−50.07
average		4.40	−0.10	−49.76
Pan *et al.*	container (qcif)	3.69	−0.23	−56.36
	news (qcif)	3.90	−0.29	−55.34
	coastguard (qcif)	2.36	−0.11	−55.03
	bus (cif)	3.85	−0.10	−58.12
	tempete (cif)	3.51	−0.23	−57.70
average		3.46	−0.19	−56.51

Table 1. Experimental results with `Intra8x8` disabled and only Intra frames.



(a) `news` QCIF $\overline{M} = 5$ (b) `news` QCIF $\overline{M} = 6$



(c) `news` QCIF $\overline{M} = 7$ (d) `silent` QCIF $\overline{M} = 5$

Fig. 8. PSNR vs. rate for only Intra coded sequences with different target number $\overline{M}$ of candidates (`Intra4x4` mode only).

ing from its average with $\overline{M} = 7$ and $T = 2$ is 0.79% while it is equal to 5.81% for the algorithm of Pan *et al.* and 30.38% for the solution proposed by Yong-dong et al. (2004).

Moreover, it is possible to tune the parameters of the fast estimation algorithm in order to vary the required computational complexity and the rate-distortion performance. The results reported in Table 1 show that reducing the parameter $\overline{M}$ by 1 permits a decrement of the computational complexity between 4.44% and 8.37%.

In addition, Table 3 reports some experimental results obtained enabling the `Intra8x8` coding mode too. In this case the average performance does not significantly change, but the complexity reduction results slightly more variable because of the increased number of coding modes. Note also that the computational saving increases since the rate-distortion optimization process becomes more complex as the mode `Intra8x8` is added, and therefore, the adoption of fast method for Intra prediction proves to be an effective strategy in the coding process.

Algorithm	**E** [**Δ**Time (%)]	**range for ΔTime (%)**
BP $\overline{M} = 5$ $T = 2$	-71.23	$[-72.44, -69.42]$
BP $\overline{M} = 6$ $T = 3$	-61.50	$[-63.64, -59.95]$
BP $\overline{M} = 7$ $T = 2$	-58.42	$[-59.21, -57.85]$
Pan *et al.* Pan et al. (2005)	-59.57	$[-65.38, -55.03]$
Yong-dong *et al.* Yong-dong et al. (2004)	-60.38	$[-68.70, -40.30]$

Table 2. Experimental results of different algorithms for only Intra mode.

$(\overline{\mathbf{M}}, \mathbf{T})$	**Sequence**	**Δ Bits (%)**	**Δ PSNR**	**Δ Time (%)**
$(6, 3)$	container (qcif)	5.93	-0.14	-66.97
	news (qcif)	6.40	-0.19	-64.63
	coastguard (qcif)	5.09	-0.18	-66.51
	bus (cif)	3.42	-0.20	-63.69
	tempete (cif)	5.66	-0.22	-61.78
	average	5.30	-0.18	-64.72
$(7, 2)$	container (qcif)	4.27	-0.13	-62.76
	news (qcif)	5.34	-0.17	-59.49
	coastguard (qcif)	3.40	-0.16	-62.37
	bus (cif)	3.12	-0.17	-59.73
	tempete (cif)	4.99	-0.20	-58.02
	average	4.23	-0.17	-60.47

Table 3. Experimental results with `Intra8x8` enabled and only Intra frames.

Final tests were devoted to evaluate the impact of the BP-based fast Intra prediction on the complexity of the overall coding process. To this purpose, the performance of the proposed fast intra algorithm was evaluated enabling Inter coding modes. In this case, the Intra coding is applied while coding Intra frames in order to find which is the best coding mode for the current macroblock (see the rate-distortion optimization routine of H.264/AVC by Joint Video Team (2004)). Table 4 reports the coding results for GOP of 100 frames with structure IP. . .P. It is possible to notice that the proposed method improves the results of the algorithm by Pan *et al.* both in terms of rate-distortion performance (we obtained lower rate increment and quality decrement for $\overline{M} = 6$) and of complexity reduction (the proposed approach permits an average 25.88% reduction in the coding time with respect to the 23.57% reduction of the algorithm by Pan et al. (2005)). For the sake of completeness, Table 5 reports the results for GOP IP. . .P and `Intra8x8` mode enabled. In this case the reduction of computational time saving approximately varies from 6.25% to 7.4% with respect to the approach with `Intra8x8` disabled (see Table 4) since the proposed algorithm significantly mitigates the computational load of the rate-distortion optimization routine (compare the results for $\overline{M} = 6$ and $T = 3$).

6. Conclusions

The chapter has described the block-based spatial prediction strategy adopted within the H.2634/AVC FRExt standard and stated the problem of enabling a low-complexity Intra coding on mobile devices. An overview of different techniques has been presented underlying the characteristics of each solution and the required complexity. Since most of the proposed solutions permit obtaining a varying computational savings which depends on the characteristics of the coded signal, the focus is centered on finding a fast Intra coding strategy that

$(\overline{\mathbf{M}}, \mathbf{T})$	**Sequence**	**Δ Bits (%)**	**Δ PSNR**	**Δ Time (%)**
$(6, 3)$	container (qcif)	1.19	−0.04	−25.00
	news (qcif)	1.00	−0.05	−25.70
	coastguard (qcif)	0.03	−0.01	−27.00
	bus (cif)	0.23	−0.01	−26.88
	tempete (cif)	0.44	−0.01	−23.82
average		0.58	−0.02	−25.68
$(7, 2)$	container (qcif)	0.74	−0.02	−18.67
	news (qcif)	0.79	−0.06	−19.14
	coastguard (qcif)	0.03	−0.01	−20.26
	bus (cif)	0.11	−0.00	−23.16
	tempete (cif)	0.38	−0.01	−22.20
average		0.41	−0.02	−20.69
Pan *et al.*	container (qcif)	1.80	−0.08	−20.78
	news (qcif)	1.23	−0.07	−23.11
	coastguard (qcif)	0.50	−0.02	−21.20
	bus (cif)	0.32	−0.01	−26.05
	tempete (cif)	0.81	−0.03	−26.72
average		0.93	−0.04	−23.57

Table 4. Experimental results with `Intra8x8` disabled GOP IP. . .P.

permits controlling the amount of tested modes (and, as a consequence, the required amount of calculation). The chapter presents a coding strategy that identifies a set of probable candidates calculating their best-mode probability. The probability estimates are obtained via Belief Propagation strategy that relies on the statistical dependence existing between spatially neighboring blocks. At the same time, the presented algorithm tries to identify the macroblock partitioning mode that better suits the current macroblock according to the coding results of the `Intra4x4` mode. Experimental results compare different algorithms and show that the Belief Propagation based strategy obtains a significant saving in terms of coding time (approximately 62%) with a negligible decrement of the PSNR value and a small average increment (less than 5.14%) in the bit rate. Moreover, the presented strategy permits an accurate control on the encoding complexity, which does not significantly vary depending on the input

$(\overline{\mathbf{M}}, \mathbf{T})$	**Sequence**	**Δ Bits (%)**	**Δ PSNR**	**Δ Time (%)**
$(6, 3)$	container (qcif)	1.25	−0.04	−31.59
	news (qcif)	1.05	−0.07	−30.78
	coastguard (qcif)	0.22	−0.01	−32.10
	bus (cif)	0.25	−0.01	−33.70
	container (qcif)	1.25	−0.04	−31.59
average		0.80	−0.03	−31.95
$(7, 2)$	container (qcif)	0.72	−0.04	−27.64
	news (qcif)	0.97	−0.07	−27.32
	coastguard (qcif)	0.15	−0.01	−27.95
	bus (cif)	0.17	−0.01	−30.02
	container (qcif)	0.72	−0.04	−27.64
average		0.55	−0.04	−28.11

Table 5. Experimental results with `Intra8x8` enabled GOP IP. . .P.

video sequence and can be tuned according to the power supply level and to the available computational resources.

Acknowledgements

Part of the work was done in collaboration with Luca Celetto of STMicroelectronics, Agrate Brianza (MI), Italy.

7. References

J. Bialkowski, A. Kaup, and K. Illgner, "Fast transcoding of intra frames between H.263 and H.264," in *Proc. of the 2004 IEEE International Conference on Image Processing (ICIP 2004)*, Singapore, Oct. 24 – 27, 2004, pp. 1151–1154.

L. Cappellari and G. A. Mian, "Analysis of joint predictive-transform coding for still image compression," *Signal Processing*, vol. 84, no. 11, pp. 2097–2114, Nov. 2004.

S. Cho, Z. Bojković, D. Milovanović, J. Lee, and J.-J. Hwang, "Image quality evaluation: Jpeg 2000 versus intra-only h.264/avc high profile," *Facta universitatis - series: Electronics and Energetics*, vol. 20, no. 1, pp. 71 – 84, Apr. 2007.

A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA, USA: Kluwer Academic Publisher, 1991.

ISO/IEC JTC1, "Coding of Audio-Visual Objects - Part 2: Visual," *ISO/IEC 14 496-2 (MPEG-4 Visual version 1), Apr. 1999; Amendment 1 (version 2), Feb. 2000; Amendment 4 (streaming profile), Jan. 2001*, Jan. 2001.

ITU-T, "Video Coding for Low Bitrate Communications, Version 1," ITU-T Recommendation H.263, 1995.

ITU-T and ISO/IEC JTC1, "Generic Coding of Moving Pictures and Associated Audio Information-Part 2: Video," ITU-T Recommendation H.262-ISO/IEC 13 818-2 (MPEG-2), 1994.

J. Jeong and D. N. Kwon, "DCT Based Fast 4X4 Intra-Prediction Mode Selection," in *Proc. of* 4th *IEEE Consumer Communications and Networking Conference (CCNC 2007)*, Las Vegas, NV, Jan. 11 – 13, 2007, pp. 332 – 335.

Joint Video Team, "Version 3 of H.264/AVC," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6),* 12th *Meeting*, Redmond, WA, USA, Jul. 17 – 23, 2004.

——, "Joint final committee draft (JFCD) of joint video specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6),* 4th *Meeting*, Klagenfurt, Germany, Jul. 2002. [Online]. Available: ftp://ftp.imtc-files.org/jvt-experts/2002_07_Klagenfurt/{JVT-D157.zip%}

H. Kalva and L. Christodoulou, "Using Machine Learning for Fast Intra MB Coding in H.264," in *Proc. of the SPIE-VCIP 2007*, San Jose, CA, USA, Feb. 2007, pp. 65 082U–1 – 65 082U–4.

C. Kim, H.-H. Shih, and C.-C. J. Kuo, "Fast H.264 intra-prediction mode selection using joint spatial and transform domain features," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 291–310, Apr. 2006.

C.-S. Kim, Q. Li, and C.-C. J. Kuo, "Fast Intra-Prediction Model Selection for H.264 Codec," in *SPIE International Symposium ITCOM 2003*, Orlando, FL, USA, Sep. 2003.

J. Kim, K. Jeon, and J. Jeong, "H.264 intra mode decision for reducing complexity using directional masks and neighboring modes," in *Proc. of* 4^{th} *IEEE Consumer Communications and Networking Conference (CCNC 2007)*, vol. 4319, Las Vegas, NV, Dec. 10 – 13, 2006, pp. 959 – 968.

F. Lorás and J.-C. Amiel, "Method for finding the prediction direction in intraframe video coding," International Patent WO 2005/088979, Paris, FR, Jan. 2005.

X. Lu and P. Yin, "Fast intra mode prediction for an H.264 encoder," Princeton, NJ, USA, Oct. 2005.

S. Milani, "A Belief-Propagation based fast Intra coding algorithm for the H.264/AVC FRExt coder," in *Proc. of the 16th European Signal Processing Conference (EUSIPCO 2008)*, Lausanne, Switzerland, Aug. 25 – 28, 2008.

F. Pan, X. Lin, S. Rahardja, K. P. Lim, and Z. G. Li, "A directional field based fast Intra mode decision algorithm for H.264 video coding," in *Proc. of 2004 IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, Jun. 27 – 30, 2004, pp. 1147–1150.

F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 813–822, Jul. 2005.

I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*. John Wiley and Sons, Sep. 2003.

J.-S. Ryu and E.-T. Kim, "Fast Intra coding method of H.264 for video surveillance system," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 7, no. 10, pp. 76–81, Oct. 2007.

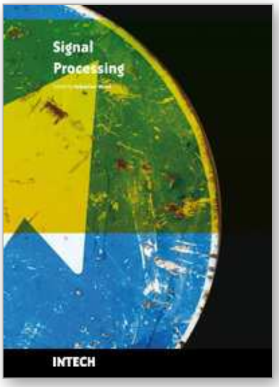
G. J. Sullivan and T. Wiegand, "Rate-Distortion Optimization for Video Compression," *IEEE Signal Processing Mag.*, pp. 74–90, Nov. 1998.

T. Wiegand and B. Girod, "Parameter Selection in Lagrangian Hybrid Video Coder Control," in *Proc. of International Conference on Image Processing, ICIP 2001*, Thessaloniki, Greece, Oct. 2001.

J. Xin and A. Vetro, "Fast mode decision for Intra-only H.264/AVC coding," in *Proc. of the 2006 Picture Coding Symposium (PCS 2006)*, Beijing, China, Apr. 24 – 26, 2006.

J. Xin, A. Vetro, and H. Sun, "Efficient macroblock coding-mode decision for H.264/AVC video coding," in *Proc. of* 24^{th} *Picture Coding Symposium (PCS 2004)*, San Francisco, CA, USA, Dec. 15 – 17, 2004.

Z. Yong-dong, D. Feng, and L. Shou-xun, "Fast 4 × 4 Intra-prediction mode selection for H.264," in *Proc. of 2004 IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, Jun. 27 – 30, 2004, pp. 1151–1154.



Signal Processing

Edited by Sebastian Miron

ISBN 978-953-7619-91-6
Hard cover, 528 pages
Publisher InTech

Published in print edition March, 2010

This book intends to provide highlights of the current research in signal processing area and to offer a snapshot of the recent advances in this field. This work is mainly destined to researchers in the signal processing related areas but it is also accessible to anyone with a scientific background desiring to have an up-to-date overview of this domain. The twenty-five chapters present methodological advances and recent applications of signal processing algorithms in various domains as telecommunications, array processing, biology, cryptography, image and speech processing. The methodologies illustrated in this book, such as sparse signal recovery, are hot topics in the signal processing community at this moment. The editor would like to thank all the authors for their excellent contributions in different areas of signal processing and hopes that this book will be of valuable help to the readers.

How to reference

Simone Milani (2010). Spatial Prediction in the H.264/AVC FRExt Coder and its Optimization, Signal Processing, Sebastian Miron (Ed.), ISBN: 978-953-7619-91-6, InTech, Available from: http://www.intechopen.com/books/signal-processing/spatial-prediction-in-the-h-264-avc-frext-coder-and-its-optimization

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821