J. Vis. Commun. Image R. 22 (2011) 687–696

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

# Novel intra prediction via position-dependent filtering

Li Zhang [a,*], Xin Zhao [b,c], Siwei Ma [a,d], Qiang Wang [e], Wen Gao [a]

[a] Institute of Digital Media, Peking University, Beijing, China
[b] Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[c] Graduate University of Chinese Academy of Sciences, Beijing, China
[d] State key laboratory of Virtual Reality Technology and Systems, Beijing University of Aeronautics and Astronautics, Beijing, China
[e] Academy of Broadcasting Science, the State Administration of Radio, Film and Television, Beijing, China

ARTICLE INFO

*Article history:*
Received 16 August 2010
Accepted 12 November 2010
Available online 17 November 2010

*Keywords:*
H.264
Image coding
Intra prediction
Interpolation filtering
Least square method
Mode-dependent directional transform (MDDT)
KTA
Singular value decomposition

ABSTRACT

This paper presents a novel intra prediction algorithm, named position-dependent filtering (PDF), to improve the intra prediction accuracy. Different from the existing schemes where the samples along one prediction direction are predicted with the same set of filtering coefficients, in the proposed PDF, position-dependent filtering coefficients are employed, i.e., different sets of filtering coefficients are pre-defined for samples with different coordinates in one coding block. For each intra prediction mode, the set of linear filtering coefficients for each position within one block is obtained from off-line training using the least square method. Moreover, to further reduce the algorithm complexity, a simplified PDF (sPDF) is proposed. In sPDF, only a subset of reference samples are used for prediction and the others are discarded because of the minor contribution to intra prediction. The proposed algorithm has been implemented in the latest ITU-T VCEG KTA software. Experimental results demonstrate that, compared with the original KTA with new intra coding tool enabled, up to 0.53 dB of average coding gain is achieved by the proposed method, while applicable computational complexity is retained for practical video codecs.



## 1. Introduction

Intra prediction (IP) is an important technique in image and video compression to exploit the spatial correlation within one picture. After intra prediction, the differences between estimated and real sample values instead of the original ones are coded and transmitted, known as residues. Good intra prediction techniques can generate residues with low energy, which is of great importance for compression. The development of intra prediction techniques has passed several decades. The basic idea can be tracked back to differential pulse code modulation (DPCM) coding in 1980 [1]. In MPEG-2 [2], transform-domain intra prediction is employed to handle DC coefficients with DPCM. After that, the related research is mainly focused on spatial-domain intra prediction with multiple prediction directions and mode selection [3–5]. One successful example is the intra prediction method defined in H.264/AVC [6]. In H.264/AVC, line-based intra prediction is employed, where the prediction block is created by extrapolating the reconstructed samples surrounding the target block along a specific direction. Moreover, to better capture the local properties of video signal, H.264/AVC employs flexible macroblock partition modes for Intra coding, including $4 \times 4$ (Intra$4 \times 4$), $8 \times 8$ (Intra$8 \times 8$) and $16 \times 16$ (Intra$16 \times 16$). For predicting the luminance component, nine prediction modes (i.e., eight directional modes plus one DC mode) are employed in both Intra$4 \times 4$ and Intra$8 \times 8$ modes, and four prediction modes (vertical, horizontal, DC and plane modes) are utilized for Intra$16 \times 16$. The efficiency of each partition mode is first evaluated by the encoder using a certain criteria, and the one which optimizes the criteria will then be selected for the actual coding.

The recent advances of intra prediction methods for further coding gain in the literature mainly concentrate on two aspects: (1) better utilization of neighboring samples and (2) utilization of global instead of local signal information. For the first aspect, efforts are devoted to investigate more intra prediction modes or better filtering methods. To capture the local information of neighboring reconstructed samples more accurately, 34 prediction modes are employed in angular intra prediction for Intra$8 \times 8$ [7], and arbitrary directional intra (ADI) for Intra$16 \times 16$ [8]. The idea of ADI is similar to angular intra prediction except for the different definition of intra prediction direction. In angular intra prediction, the prediction direction is given by the displacement of the bottom row/rightmost column of the block and the reference row/column around the block. While in ADI, the intra prediction direction is determined by a tuple $(dx, dy)$ which defines the line to be used to calculate the reference samples of the current pixel. Constructing several intra prediction subsets by uniform rotation from

* Corresponding author.
*E-mail address:* li.zhang@pku.edu.cn (L. Zhang).

1047-3203/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2010.11.003

original predictors of H.264/AVC is introduced in [9]. In [10], bidirectional intra prediction is proposed to combine two kinds of the existing intra prediction modes. To get better interpolation filtering coefficients, the available reconstructed region within a template is used to calculate the filtering coefficients [11]. Liu et al. utilized two windows of reconstructed pixels from two previous frames to train the linear prediction coefficients at both encoder and decoder [12]. In [13], Wang proposed a distance-based weighted prediction method to improve the prediction efficiency of DC mode. Later, a weighted cross prediction mode is proposed [14]. With the revised interpolation method, the decorrelation ability is enhanced and higher coding gain can be achieved.

For some usual cases, the image blocks have repeated patterns instead of distinctive direction information. In this case, utilizing the global information instead of the spatial neighboring samples will bring in better coding efficiency. Related research works include intra displacement vector and template matching, etc. [15–20]. In [15], intra displacement compensation (IDC) utilizes an intra displacement vector per block or partition to get the reference samples. Other related works including intra prediction based on template matching using a single template [16], backward-adaptive texture synthesis [17], multiple candidates [18], priority-guided template matching [19] and locally adaptive illumination compensation [20] are proposed and studied.

In our simulations, it is observed that the correlation among different target samples and reference samples shows diversity. Therefore, in this paper, a position-dependent filtering (PDF) approach is proposed to further improve the intra prediction accuracy. In PDF, each target sample is predicted by extrapolating the neighboring reconstructed ones with its own set of linear filtering coefficients. To obtain these sets of filtering coefficients, the least square method is used for off-line training. Furthermore, in consideration of computational complexity, the filtering coefficients are fixed during the encoding or decoding process. Different from the intra prediction method in H.264/AVC, the sets of linear filtering coefficients in our proposed method are designed to be both mode- and position-dependent. That is to say, for different locations of samples within one block, different sets of filtering coefficients are employed. With the proposed method, the intra prediction is further refined and therefore higher prediction accuracy is achieved. Moreover, to further reduce the algorithm complexity, a simplified PDF (sPDF) is proposed. In sPDF, only a subset of reference samples are used for prediction and others are discarded because of their minor contribution to intra prediction.

The remainder of this paper is organized as follows. In Section 2, a brief review of the intra prediction method in H.264/AVC is provided. The proposed PDF are discussed in Section 3, with detailed descriptions including the motivations, PDF algorithm, filtering coefficient matrix derivation process, implementation and simplification. Performance of the proposed method is validated in Section 4 with extensive experimental results and analysis. Finally, in Section 5, the paper is concluded and some expected future work is also presented.

## 2. Spatial intra prediction in H.264/AVC

In H.264/AVC, for Intra4 $\times$ 4 and Intra8 $\times$ 8 macroblock partitions, nine intra prediction modes are available including one DC mode and eight directional modes. For example, the directional modes in Intra4 $\times$ 4 are illustrated in Fig. 1(a). The prediction block is derived by extrapolating the neighboring reconstructed samples along some direction with fixed coefficients. Fig. 1(b) illustrates 16 samples of one $4 \times 4$ block which are predicted by the reconstructed samples of upper and left-hand reference samples of current block. The filtering process to obtain the predictor values is not shown in this figure. In Fig. 1(b), the samples to be predicted in one $4 \times 4$ block and their reference samples are labeled as $C_i$ $(0 \leqslant i \leqslant 15)$ and $P_j$ $(0 \leqslant j \leqslant 12)$, respectively. For brevity, let $\vec{P}$ denote a vector containing the reconstructed reference sample values (e.g., $P_0$ to $P_{12}$ form a $13 \times 1$ column vector $\vec{P}$). Let $\vec{\hat{C}}^k$ denote a vector containing the predicted sample values of the current coded block for a specific spatial IP mode $k$ (e.g., predictor values $\vec{\hat{C}}^k$ of all samples $C_i$ form a $16 \times 1$ vector $\vec{\hat{C}}_i^k$). The filtering coefficient matrix $W^k$ corresponding to the filtering coefficient sets can be defined to specify the IP mode. Here, $W^k$ may be expressed as follows:

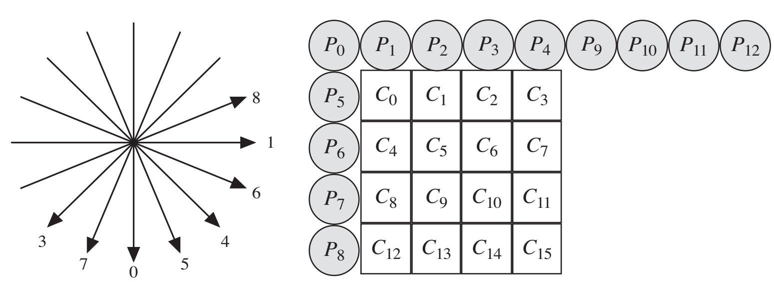
$$W^k = \begin{bmatrix} w^k_{0,0}, & w^k_{0,1}, & \dots, & w^k_{0,11}, & w^k_{0,12} \\ w^k_{1,0}, & w^k_{1,1}, & \dots, & w^k_{1,11}, & w^k_{1,12} \\ \vdots & \vdots & & \vdots & \vdots \\ w^k_{14,0}, & w^k_{14,1}, & \dots, & w^k_{14,11}, & w^k_{14,12} \\ w^k_{15,0}, & w^k_{15,1}, & \dots, & w^k_{15,11}, & w^k_{15,12} \end{bmatrix}. \tag{1}$$

By doing the following multiplication, we can obtain the predicted value vector $\vec{\hat{C}}^k$ as follows:

$$\vec{\hat{C}}^k = W^k \cdot \vec{P}. \tag{2}$$

That is,

$$\begin{bmatrix} \hat{C}^k_0 \\ \hat{C}^k_1 \\ \vdots \\ \hat{C}^k_{14} \\ \hat{C}^k_{15} \end{bmatrix} = \begin{bmatrix} w^k_{0,0}, & w^k_{0,1}, & \dots, & w^k_{0,11}, & w^k_{0,12} \\ w^k_{1,0}, & w^k_{1,1}, & \dots, & w^k_{1,11}, & w^k_{1,12} \\ \vdots & \vdots & & \vdots & \vdots \\ w^k_{14,0}, & w^k_{14,1}, & \dots, & w^k_{14,11}, & w^k_{14,12} \\ w^k_{15,0}, & w^k_{15,1}, & \dots, & w^k_{15,11}, & w^k_{15,12} \end{bmatrix} \cdot \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_{11} \\ P_{12} \end{bmatrix}. \tag{3}$$




(a) Directional intra modes (b) Boundary and inside samples

**Fig. 1.** Spatial intra4 $\times$ 4 prediction in H.264/AVC.

Note that, in H.264/AVC, the filtering coefficients of samples along one specified direction are the same, which will result in one unique predictor values of samples along one specified direction. For those samples not along the prediction direction, different reference samples are used, but with the same filtering coefficients. Let us consider the IP mode '0', i.e., vertical mode, the row vectors of filtering coefficient matrix $W^0$ for samples $C_0$, $C_4$, $C_8$ and $C_{12}$ are the same. They are:

$$\begin{aligned} &W^0(0,:) = [0,1,0\ldots,0,0],\\ &W^0(4,:) = [0,1,0\ldots,0,0],\\ &W^0(8,:) = [0,1,0\ldots,0,0],\\ &W^0(12,:) = [0,1,0\ldots,0,0]. \end{aligned} \tag{4}$$

While for samples $C_1$, $C_5$, $C_9$ and $C_{13}$, the corresponding filtering coefficient vectors are:

$$\begin{aligned} &W^0(1,:) = [0,0,1,0,\ldots,0,0],\\ &W^0(5,:) = [0,0,1,0,\ldots,0,0],\\ &W^0(9,:) = [0,0,1,0\ldots,0,0],\\ &W^0(13,:) = [0,0,1,0,\ldots,0,0]. \end{aligned} \tag{5}$$

For other intra prediction modes, the same phenomenon can also be observed. Thus, this kind of line-based intra prediction can be viewed as mode-dependent, but can not be treated as totally position-dependent.

Although the intra prediction in H.264/AVC shows lower complexity and good coding efficiency, there is still room for further improvement. As pointed out in [21], the correlation between two samples is inversely relative to the distance between them. Therefore, samples at each position within one block should employ different linear filtering coefficients to derive the predictor. Intuitively, it is worthwhile to design the prediction filtering coefficients taking both mode and position into consideration.

## 3. Enhanced intra prediction via position-dependent filtering

In this section, several experimental observations which motivate the basic idea of the proposed PDF will be discussed first. Then the proposed method will be introduced in detail, including filtering coefficient matrix derivation process and implementation issues. Finally, to further reduce the computational complexity for practical applications, a simplified PDF algorithm is introduced with revised efficient reference samples and shorter filter taps, while maintaining the original coding performance very well.

### 3.1. Motivation and observations

The line-based intra prediction method in H.264/AVC efficiently improves the Intra coding efficiency by extrapolating the reconstructed samples with variable filtering coefficients for different IP modes. However, in our simulations, it is observed that even with one IP mode, the samples to be predicted along the direction always exhibit various correlations with their neighboring reference samples. To verify this conclusion, the following experiments have been conducted. Firstly, we choose four frames from each of eight sequences at 1920 × 1080 resolution to construct a new test sequence. In this way, the diverse statistical characteristics of natural images can be approximately covered. Secondly, these 32 frames are coded as intra frames using H.264/AVC. Thirdly, based on the rate-distortion (R-D) optimization results, the coordinates of blocks coded in the same IP mode are marked and the corresponding samples of the original input sequence are selected for calculating the correlation coefficients. The correlation coefficient is defined as follows:

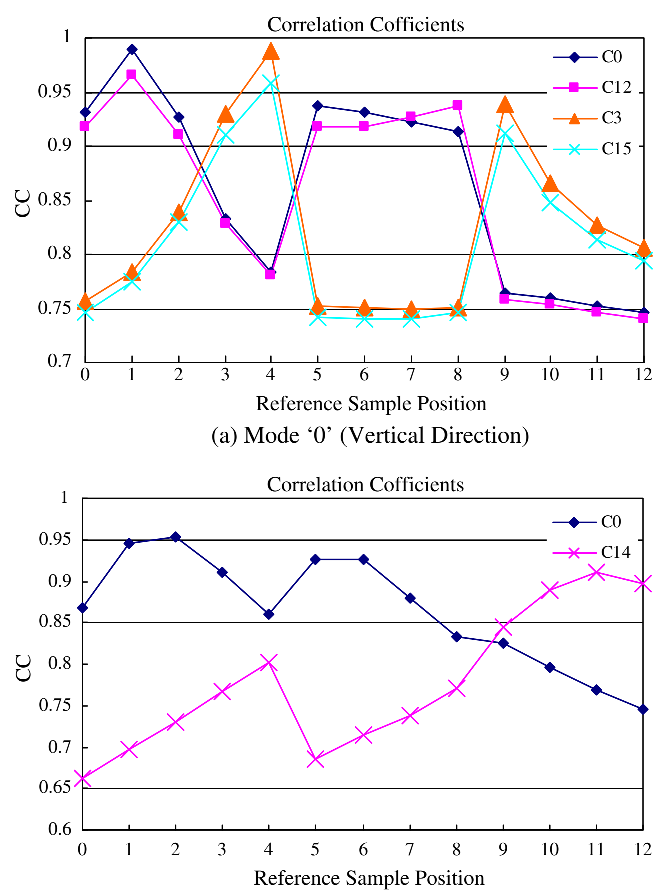
$$r_{C_i,P_j} = \frac{N\sum_{n=1}^{N} C_{i,n}P_{j,n} - \sum_{n=1}^{N} C_{i,n}\sum_{n=1}^{N} P_{j,n}}{\sqrt{N\sum_{n=1}^{N} C_{i,n}^2 - \left(\sum_{n=1}^{N} C_{i,n}\right)^2}\sqrt{N\sum_{n=1}^{N} P_{j,n}^2 - \left(\sum_{n=1}^{N} P_{j,n}\right)^2}}, \tag{6}$$

where $r_{C_i,P_j}$ is the correlation coefficient between the $i$th sample to be predicted and its $j$th reference sample. $C_{i,n}$ denotes the $i$th sample value within the $n$th block. $P_{j,n}$ represents the $j$th reference sample of the $n$th block. $N$ is the number of blocks coded with the same IP mode. Fig. 2 depicts the correlation coefficients among samples $C_0$, $C_3$, $C_{12}$, $C_{14}$, $C_{15}$ and their 13 neighboring reference samples. The correlation coefficients in Fig. 2(a) and (b) are derived with blocks coded with vertical and diagonal down left directions, respectively. From these figures, it is not hard to observe that the correlation between $C_i$ and $P_j$ is variable, even those along the same texture direction, such as $C_3$ and $C_{15}$ in Fig. 2(a). Another observation is that each sample may have several high-correlated reference samples. Take sample $C_0$ (Mode '3') for example, the correlation coefficients between $C_0$ and $P_1$, $P_2$, $P_3$, $P_5$ and $P_6$ are all larger than 0.9.

The above observations imply the possibility to further improve the intra prediction efficiency with revised filtering process, and it leads us to the idea of using position-dependent filtering method which is naturally capable to enhance the decorrelation ability for further coding gain.

### 3.2. Intra prediction via position-dependent filtering

To maintain applicable computational complexity for practical video codecs, PDF employs the conventional line-based intra prediction method with fixed interpolation filtering. Each sample within a target sub-block is also predicted as a linear weighted




(a) Mode '0' (Vertical Direction)

(b) Mode '3' (Diagonal Down Left Direction)

**Fig. 2.** Examples of correlation coefficients of Intra4 × 4.

sum of the reconstructed samples around the target block. However, there are two important differences between PDF and the conventional method:

(1) PDF employs different sets of filtering coefficients for samples located at different position within one coding block, even for those along the IP direction. The derivation process of filtering coefficients will be described in the following subsection.
(2) In our proposed PDF, all the samples within one block use all of the neighboring reconstructed samples around the target block. Samples located at each position within one block utilize the same set of reference samples. To illustrate this key difference between the conventional IP method in H.264/AVC and PDF, the reference samples of both algorithms have been depicted in Fig. 3. As it is shown in Fig. 3(a), for IP mode '1', the reference sample of sample $C_{11}$ is specified as $P_7$ in H.264/AVC. However, in the proposed PDF, there is a group of reference samples, $P_j$ ($j = 0, 1, \ldots, 8$) for predicting the target sample, shown in Fig. 3(b).

In summary, in our proposed PDF, for IP mode $k$, the predictor of the $i$th target sample $C_i$ ($0 \leqslant i \leqslant 15$ for Intra4 × 4; $0 \leqslant i \leqslant 63$ for Intra8 × 8) can be obtained from its neighboring samples as:

$$\widehat{C}_i^k = \sum_{j=0}^{N-1} W^k(i,j) \times P_j, \tag{7}$$

where $k$ denotes the IP mode index. $P_j$ ($0 \leqslant j < N$) represents the $j$th neighboring reconstructed intensity value of $C_i$. $W^k(i,j)$ is the derived linear prediction coefficient for the ($C_i$, $P_j$) pair under IP mode $k$. $N$ represents the filter tap, that is the number of neighboring samples utilized for prediction. To take full advantage of the local information, the filter tap is defined as follows.

For Intra4 × 4 modes:

$$N = \begin{cases} 13, & \text{diagonal down left and vertical left IP modes} \\ 9, & \text{otherwise}. \end{cases} \tag{8}$$

For Intra8 × 8 modes:

$$N = \begin{cases} 25, & \text{diagonal down left and vertical left IP modes,} \\ 17, & \text{otherwise}. \end{cases} \tag{9}$$

The corresponding neighboring samples for prediction are those within the left column and the above row of the target block, as depicted in Fig. 1(b) for Intra4 × 4 and Fig. 4 for Intra8 × 8, respectively. In this way, the nearest reconstructed samples can be utilized for achieving better prediction accuracy. Note that the linear coefficient is mode and position dependent. Samples at different positions within one block or under different IP mode employ different linear filtering coefficients. Therefore, the proposed PDF is able to better reduce the spatial redundancies among the target samples and the reference samples, which can bring in higher prediction accuracy and further coding gain.

### 3.3. Filtering coefficient matrix derivation process

Linear filtering coefficients can be derived through different ways. For example, we can apply either the average or median filters to the neighboring samples. These methods can be easily implemented but have severe sacrifice of prediction accuracy. Naturally, the more accurate the linear filtering coefficients are, the higher the coding efficiency will be. To accurately derive the filtering coefficient matrix $W^k$, the least square method is employed for the off-line training process. The derivation process is described as follows.

Firstly, a training sequence is coded with the conventional intra prediction method. To better cover the characteristics of intra-predicted residual in natural images, eight sequences in 1920 × 1080 format are utilized to construct a training sequence. The first four frames of each test sequence are extracted to construct the training sequence with total 32 frames. The training sequence is then intra coded and the rate-distortion optimized mode of each sub-block is recorded for the following samples classification. To derive the linear filtering coefficients of both Intra4 × 4 and Intra8 × 8, adaptive block-size transform is also used in our training process.

Secondly, training sample databases are set up for each position and each prediction mode with sample classification. Sub-blocks with the same IP mode based on the rate-distortion decision will be treated as one category. Samples located in one category at the original frame construct one training database. Note that we use the original samples, including target sample $C_i$ and its neighboring reference sample $P_j$, instead of the reconstructed ones in order to obtain more accurate filtering coefficients. To calculate $W^k(i,j)$ for each position, we select only valid samples from training databases. Valid samples should satisfy two more conditions besides the same IP mode:

(1) They have the same coordinate relative to the sub-block to make sure the linear prediction is position dependent.
(2) For diagonal down left and vertical left IP modes, the left, up and up-right blocks should exit; for other IP modes; all the samples within the left column and the above row of the coding block must exist.

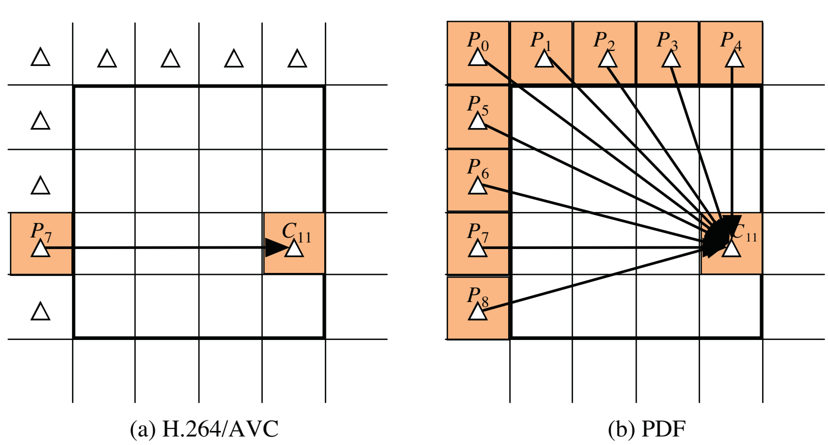



**Fig. 3.** Comparison of reference samples of horizontal mode in Intra4 × 4.

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $P_0$ | $P_1$ | $P_2$ | $P_3$ | $P_4$ | $P_5$ | $P_6$ | $P_7$ | $P_8$ | $P_{17}$ | $P_{18}$ | $P_{19}$ | $P_{20}$ | $P_{21}$ | $P_{22}$ | $P_{23}$ | $P_{24}$ |
| $P_9$ | $C_0$ | $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ | $C_6$ | $C_7$ | | | | | | | | |
| $P_{10}$ | $C_8$ | $C_9$ | $C_{10}$ | $C_{11}$ | $C_{12}$ | $C_{13}$ | $C_{14}$ | $C_{15}$ | | | | | | | | |
| $P_{11}$ | $C_{16}$ | $C_{17}$ | $C_{18}$ | $C_{19}$ | $C_{20}$ | $C_{21}$ | $C_{22}$ | $C_{23}$ | | | | | | | | |
| $P_{12}$ | $C_{24}$ | $C_{25}$ | $C_{26}$ | $C_{27}$ | $C_{28}$ | $C_{29}$ | $C_{30}$ | $C_{31}$ | | | | | | | | |
| $P_{13}$ | $C_{32}$ | $C_{33}$ | $C_{34}$ | $C_{35}$ | $C_{36}$ | $C_{37}$ | $C_{38}$ | $C_{39}$ | | | | | | | | |
| $P_{14}$ | $C_{40}$ | $C_{41}$ | $C_{42}$ | $C_{43}$ | $C_{44}$ | $C_{45}$ | $C_{46}$ | $C_{47}$ | | | | | | | | |
| $P_{15}$ | $C_{48}$ | $C_{49}$ | $C_{50}$ | $C_{51}$ | $C_{52}$ | $C_{53}$ | $C_{54}$ | $C_{55}$ | | | | | | | | |
| $P_{16}$ | $C_{56}$ | $C_{57}$ | $C_{58}$ | $C_{59}$ | $C_{60}$ | $C_{61}$ | $C_{62}$ | $C_{63}$ | | | | | | | | |

**Fig. 4.** Reference sample definitions in of Intra8 × 8 PDF.

Thirdly, the linear filtering coefficients are derived with the least square method for each IP mode and each position within one block. For all the valid training samples, we approximate them as the linear weighted sum of the neighboring samples. The approximation process has been defined in formula (7). The distortion between the actual and the approximated value of the valid training sample $C_i$ under IP mode $k$ can be computed as:

$$D(C_i^k) = \left(C_i^k - \widehat{C}_i^k\right)^2, \tag{10}$$

where $C_i^k$ denotes the actual value of sample $C_i$ under IP mode $k$ and $\widehat{\vec{C}}_i^k$ represents the approximated values obtained by formula (7). Actually, $C_i^k$ equals to $C_i$ under every intra mode. Based on the above definitions, the optimal filtering coefficients for intra mode $k$ and position $i$ correspond to those which minimize the distortion between all the actual valid training sample values and the approximated ones, which can be expressed as:

$$\vec{W}^k_{optimal}(i,:) = \underset{\vec{W}^k(i,:)}{argmin} \sum_{s_m \in S_{valid}} D(s_m), \tag{11}$$

where $S_{valid}$ represents the set of all the valid training samples and $s_m$ represents the $m$th sample within $S_{valid}$. $\vec{W}^k_{optimal}(i,:)$ is the derived $1 \times N$ row vector for IP mode $k$ and position $i$. All the row vectors $\vec{W}^k_{optimal}(i,:)$ ($0 \leqslant i \leqslant 15$ for Intra4 × 4; $0 \leqslant i \leqslant 63$ for Intra8 × 8) form the filtering coefficient matrix $W^k$. According to the least square method, the optimal prediction coefficient vector $\vec{W}^k_{optimal}(i,:)$ can be derived as:

$$\vec{W}^k_{optimal}(i,:) = (P^TP)^{-1}(P^T\vec{S}), \tag{12}$$

where $\vec{S}$ denotes a column vector of valid training samples with length of $M$. $P$ denotes the neighboring sample matrix for each pixel in $\vec{S}$ and it is a $M \times N$ matrix, with $M$ representing the number of valid training samples and $N$ representing the number of reference samples for each training sample. Fig. 5 gives an example of the calculated filtering coefficient matrix for Intra4 × 4 mode '0'. From this figure and the correlation coefficients depicted in Fig. 2(a), it can be observed that the magnitude information of filtering coefficients is consistent with the correlation strength. That is to say, if the $j$th reference sample presents higher correlation with sample $C_i$ in IP mode $k$, then the magnitude of the corresponding filtering coefficient $W^k(i,j)$ will be larger.

With the above three process modules, the linear filtering coefficients for each intra mode and each relative position are obtained. The derived filtering coefficients are used in the intra prediction process of our proposed method, and kept as constants throughout the encoding or decoding process.

$$W^0 = \begin{bmatrix}
-0.3991, & 0.8886, & 0.0536, & -0.0148, & 0.0011, & 0.4242, & 0.0898, & -0.0421, & -0.0016 \\
-0.0906, & 0.0227, & 0.9403, & 0.0380, & -0.0059, & 0.0588, & 0.0543, & -0.0204, & 0.0032 \\
-0.0505, & -0.0251, & 0.0585, & 0.9217, & 0.0366, & 0.0465, & 0.0103, & 0.0024, & -0.0012 \\
-0.0395, & -0.0155, & 0.0247, & 0.0010, & 0.9803, & 0.0296, & 0.0128, & 0.0085, & -0.0024 \\
-0.3299, & 0.8186, & 0.0691, & -0.0167, & 0.0018, & -0.0079, & 0.4571, & 0.0411, & -0.0341 \\
-0.0724, & 0.0393, & 0.8995, & 0.0492, & -0.0024, & -0.0093, & 0.0734, & 0.0450, & -0.0225 \\
-0.0205, & -0.0368, & 0.1016, & 0.8605, & 0.0566, & -0.0249, & 0.0467, & 0.0288, & -0.0143 \\
-0.0244, & -0.0216, & 0.0404, & 0.0084, & 0.9535, & -0.0238, & 0.0402, & 0.0176, & 0.0077 \\
-0.2689, & 0.7753, & 0.0668, & -0.0076, & -0.0040, & -0.0426, & -0.0124, & 0.4467, & 0.0452 \\
-0.0487, & 0.0544, & 0.8575, & 0.0568, & 0.0022, & -0.0372, & -0.0041, & 0.0591, & 0.0594 \\
-0.0034, & -0.0396, & 0.1258, & 0.8111, & 0.0738, & -0.0361, & -0.0009, & 0.0261, & 0.0399 \\
-0.0135, & -0.0266, & 0.0522, & 0.0172, & 0.9267, & -0.0349, & 0.0100, & -0.0040, & 0.0700 \\
-0.2386, & 0.7200, & 0.0640, & 0.0057, & -0.0021, & -0.0277, & -0.0173, & -0.0897, & 0.5837 \\
-0.0323, & 0.0679, & 0.7961, & 0.0675, & 0.0184, & -0.0256, & 0.0031, & -0.1162, & 0.2192 \\
0.0092, & -0.0311, & 0.1443, & 0.7475, & 0.0994, & -0.0309, & 0.0175, & -0.0949, & 0.1337 \\
-0.0042, & -0.0254, & 0.0724, & 0.0176, & 0.8916, & -0.0442, & 0.0279, & -0.0692, & 0.1289
\end{bmatrix}$$

**Fig. 5.** Filtering coefficient matrix for Intra4 × 4 mode '0' ($N$ = 9).

**Table 1**
Test sequences in the experiments.

| Resolution | Frame rate (HZ) | Sequences |
|---|---|---|
| QCIF | 15 | Bus, Soccer, Tempete, Harbour, City |
| CIF | 30 | Kiel, City, Harbour, Flower, Stefan, Bus, Tempete |
| 720p | 60 | City, Night, Harbour, Raven |
| 1080p | 24 | ParkScene |
| | 50 | Crowdrun, Parkjoy, Wisley2 |
| 1600p | 30 | PeopleOnStreet, Traffic |

### 3.4. Implementation issues

To efficiently integrate our proposed PDF into an intra coding system, there are two more issues that need to be considered. The first one is to avoid floating-point operations. The linear filtering coefficients obtained from the training process using the least square method are originally floating-point. To implement the proposed method with fixed-point operation, $W^k(i,j)$ is first scaled to a 16-bit integer. Formula (7) can then be rewritten as:

$$\widehat{C}_i^k = \left(\sum_{j=0}^{N-1} W^k(i,j) \times P_j + (1 \ll 15)\right) \gg 16. \qquad (13)$$

In our simulations, it is observed that 16-bit integer precision is good enough to keep the coding efficiency. Furthermore, after using PDF instead of the one defined in H.264/AVC, the statistical characteristics of prediction residues after PDF have been changed. Then, it leads to the other implementation issue: how to design the associated transforms to better remove the remaining correlation among prediction residues. To solve this problem, we update the transform matrices for each IP mode of Intra4 × 4 and Intra8 × 8


(a) Bus (b) Tempete (c) Soccer
(d) City (e) Harbour (f) Kiel
(g) Flower (h) Stefan
(i) Night (j) Raven

**Fig. 6.** Test sequences.

as the mode-dependent directional transform (MDDT) technique does [22]. In the transform matrix design process, the singular value decomposition (SVD) method is utilized for off-line training.

### 3.5. Simplified PDF

Although the aforementioned PDF method achieves superior coding performance over conventional intra prediction method, the computational complexity of both encoder and decoder is also increased due to the longer filter tap. To further reduce the computational complexity and make it more suitable for practical applications, in this subsection, we will present the design of the simplified PDF (sPDF).

The basic idea of the proposed sPDF is to choose relatively more important reference samples and reduce the filter tap with the aid of the correlation strength. Since the magnitudes of filtering coefficients have a good match with the correlation strength, we can choose the reference samples for each position directly based on the magnitudes of filtering coefficients. The reference samples of sPDF can be obtained with the following procedure:

$$\vec{P}^{k,i}_{N_s\times 1} = [P_{l_0}, P_{l_1}, \ldots, P_{l_{N_s-1}}]^T, \tag{14}$$

where $N_s$ denotes the reduced filter tap length for IP mode $k$ and position $i$. $T$ represents the transposition operation and $l_k$ ($0 \leqslant k < N_s$) is defined as the position index that the corresponding reference sample owns the $k$th largest magnitude of the filtering coefficient. After this selection, the reference sample set for each position may be different. Therefore, the predictor derivation process defined in formula (7) for PDF is unsuitable for sPDF. In sPDF, the predictor should be calculated as:

$$\widehat{C}_i^k = \sum_{j=0}^{N_s-1} W^k(i,j) \times P_j^{k,i}. \tag{15}$$

In our simulation, the reduced filter tap $N_s$ is defined as follows.

For Intra4 × 4 modes:

$$N_s = \begin{cases} 9, & \text{Diagonal down left and Vertical left IP modes,} \\ 7, & \text{otherwise} \end{cases} \tag{16}$$

while for Intra8 × 8 modes, $N_s$ is 13 for all IP modes. During our initial tests, such settings of $N_s$ for Intra4 × 4 and Intra8 × 8 could achieve the besting coding efficiency. Therefore, in the following experimental tests, we use these filtering tap definitions. After the reference samples and filtering tap are chosen, the associated filtering coefficient matrix will be re-calculated from off-line training using the least square method. The training process is the same as mentioned in Section 3.3.

(k) Crowdrun (l) ParkScene

(m) Parkjoy (n) Wisely2

(o) PeopleOnStreet (p) Traffic

**Fig. 6** (*continued*)

In summary, sPDF distinguishes from PDF in the following two parts:

(1) The filter taps in sPDF are shorter than those employed in PDF.
(2) In sPDF, each sample within one block owns a specific set of the reconstructed neighboring samples as reference samples. The reference sample sets are position-dependent.

## 4. Experimental results and analysis

To validate the efficiency of the proposed methods, we integrate our methods into the state-of-art KTA platform and make comparisons. Since the proposed method only focuses on intra coding, all-intra coding configuration is used in the experiments to show the advantage of our proposed methods. Moreover, evaluations and analysis over the complexity issue are also given in this section.

### 4.1. R-D performance comparison with KTA

To verify the performance of the proposed intra prediction method, the proposed methods are implemented on the latest KTA software with version 2.6 [23]. Only the nine intra prediction modes of Intra4×4 and Intra8 × 8 of the luminance component are replaced with our proposed methods. For Intra16 × 16 and chrominance component, they are kept the same as those in H.264/AVC. To validate the performance of PDF and sPDF on various video contexts and resolutions, extensive experiments have been made on a wide range of test sets including QCIF (176 × 144), CIF (352 × 288), 720p (1280 × 720), 1080p (1920 × 1080) and 1600p (2560 × 1600) formats as shown in Table 1. Individual frames of these test sequences which are widely used in video coding standardization groups and research fields are also shown in Fig. 6.

For all the test sequences, simulations are performed on the first 100 frames in each sequence, with all-Intra coding configuration. The only adopted intra coding technique, MDDT, is also enabled in our simulations to achieve the anchor data. Some other important encoding configurations are shown in Table 2 for more details. To calculate the average difference between two R-D curves, we employ the popular BD-Rate and BD-PSNR [24] for performance evaluations.

Table 3 tabulates the experimental results compared with KTA, where the coding gain is calculated including both luminance and chrominance components. From this table, it can be observed that, for Intra coding, the proposed PDF achieves about average 0.292 dB gain of BD-YPSNR, or an equivalent of 3.535% BD-Rate reduction compared with KTA. Among these 22 test sequences, the best case is 0.519 dB. When compared to PDF, sPDF can achieve almost the same coding gain. For some sequences, sPDF presents even slightly better performance than PDF. An explanation for this observation is that longer filter taps in PDF may cause an over-fitting problem, i.e., the filter coefficients are more heavily relied on the training data and becomes less robust for other sequences which are not included in the training set. For some cases, shorter filter taps could introduce more reliable filtering coefficients, and also better coding performance. For a comparison of the R-D behaviors in the entire QP range, the R-D curves of both the anchor and proposed methods for sequences *Harbour* in 720p format and *Crowdrun* in 1080p format are shown in Fig. 7(a) and (b), respectively. Here, “anchor” represents the result achieved by the default intra prediction method in KTA 2.6, “PDF” represents the result of the proposed method with original implementation, and “sPDF” indicates the result of the proposed method with simplified implementation. From these figures, it is observed that the proposed methods outperform the anchor from low to high bitrates.

**Table 2**
Encoding configurations in the experiments.

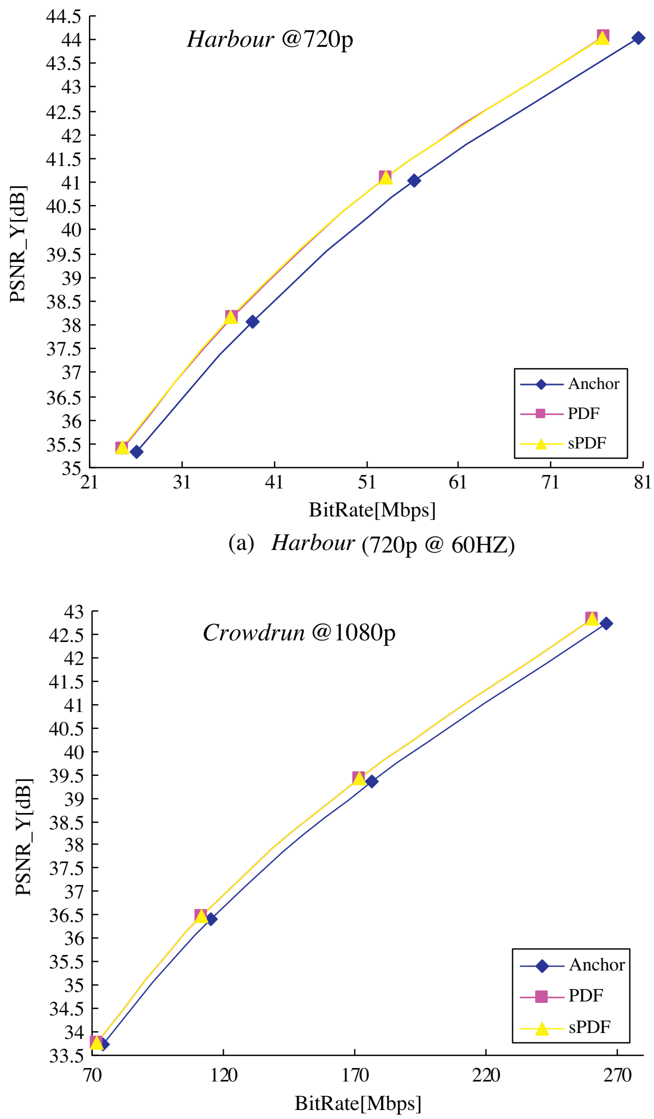
| Platform | Encoding configurations |
|---|---|
| KTA 2.6 | Rate-distortion optimization with high complexity<br>Adaptive block-size transform<br>Context-based Adaptive Binary Arithmetic Coding (CABAC)<br>MDDT<br>IntraPeriod = 1<br>FrameSkip = 0<br>NumberBFrames = 0<br>QP = {20, 24, 28, 32}<br>FramesToBeEncoded = 100 |

**Table 3**
Performance of the proposed algorithms on KTA2.6.

| Sequences | | PDF | | sPDF | |
|---|---|---|---|---|---|
| | | BD-YPSNR (dB) | BD-Rate (%) | BD-YPSNR (dB) | BD-Rate (%) |
| QCIF | Bus | 0.207 | −1.981 | 0.243 | −2.309 |
| | Soccer | 0.132 | −1.907 | 0.143 | −2.049 |
| | Tempete | 0.212 | −1.787 | 0.237 | −2.000 |
| | Harbour | 0.430 | −4.206 | 0.460 | −4.497 |
| | City | 0.289 | −3.511 | 0.308 | −3.727 |
| QCIF average | | 0.254 | −2.678 | 0.278 | −2.916 |
| CIF | Kiel | 0.180 | −1.675 | 0.210 | −1.934 |
| | City | 0.257 | −3.139 | 0.282 | −3.429 |
| | Harbour | 0.421 | −4.311 | 0.460 | −4.692 |
| | Flower | 0.352 | −2.570 | 0.397 | −2.887 |
| | Stefan | 0.232 | −2.050 | 0.269 | −2.374 |
| | Bus | 0.281 | −2.901 | 0.311 | −3.211 |
| | Tempete | 0.324 | −3.138 | 0.338 | −3.272 |
| CIF average | | 0.292 | −2.826 | 0.324 | −3.114 |
| 720p | City | 0.406 | −5.006 | 0.400 | −4.938 |
| | Night | 0.245 | −3.324 | 0.246 | −3.343 |
| | Harbour | 0.519 | −6.499 | 0.530 | −6.643 |
| | Raven | 0.263 | −4.856 | 0.257 | −4.743 |
| 720p average | | 0.358 | −4.921 | 0.358 | −4.917 |
| 1080p | Crowdrun | 0.274 | −3.810 | 0.279 | −3.880 |
| | Parkjoy | 0.296 | −3.791 | 0.305 | −3.906 |
| | ParkScene | 0.224 | −4.382 | 0.226 | −4.427 |
| | Wisley2 | 0.284 | −3.243 | 0.296 | −3.369 |
| 1080p average | | 0.270 | −3.807 | 0.277 | −3.896 |
| 1600p | PeopleOnStreet | 0.297 | −4.654 | 0.283 | −4.435 |
| | Traffic | 0.293 | −5.024 | 0.283 | −4.864 |
| 1600p average | | 0.295 | −4.839 | 0.283 | −4.650 |
| Total average | | 0.292 | −3.535 | 0.307 | −3.679 |

### 4.2. Complexity analysis

The proposed methods are compatible with the existing coding schemes. Since our proposed method does not add any additional prediction directions, no additional overhead need to be transmitted to the decoder side. At the same time, there is no difference in the R-D optimization process for selecting the best prediction mode. The main differences between the proposed methods and the conventional one in H.264/AVC are in the filtering coefficient precision and filter tap. In H.264/AVC, the elements in the filtering coefficient matrix are no larger than three. Therefore, 2-bit precision will be enough for storing filtering coefficients. However, in PDF or sPDF, the filtering coefficient is 16-bit precision. Therefore, both the encoder and the decoder need additional memories to store the filtering coefficient matrix. In the proposed sPDF, for the Diagonal_down_left and Vertical_left modes of Intra4 × 4, the filter tap is set to be 9, i.e., for each pixel in one 4 × 4 block, 9 filtering coefficients are needed, therefore, for each 4 × 4 block, 9 × 4 × 4 elements are needed to store the filtering coefficient




(a) *Harbour* (720p @ 60HZ)

(b) *Crowdrun* (1080p @ 50Hz)

**Fig. 7.** R-D performance comparisons with pure-intra frame coding.

**Table 4**
Comparisons of encoding time between the one in KTA and sPDF.

| Sequences | QP | Encoding time (s) | | |
|---|---|---|---|---|
| | | Anchor | Proposed | Proposed (optimized) |
| City (720p) | 20 | 2088.879 | 4035.065 | 1825.288 |
| | 24 | 1692.906 | 4623.974 | 1680.515 |
| | 28 | 1654.278 | 4230.143 | 1565.133 |
| | 32 | 1547.423 | 2248.755 | 1471.339 |
| Night (720p) | 20 | 2182.317 | 2575.856 | 1752.148 |
| | 24 | 1460.063 | 2393.206 | 1622.755 |
| | 28 | 1316.093 | 2242.372 | 1523.137 |
| | 32 | 1437.883 | 2127.354 | 1445.916 |
| Harbour (720p) | 20 | 1751.967 | 2640.095 | 1793.45 |
| | 24 | 1853.051 | 2444.785 | 1654.514 |
| | 28 | 1394.988 | 2302.753 | 1548.805 |
| | 32 | 1271.902 | 2184.981 | 1467.162 |
| Raven (720p) | 20 | 1339.459 | 2253.082 | 1530.64 |
| | 24 | 1217.034 | 2134.547 | 1444.542 |
| | 28 | 1134.344 | 2056.249 | 1388.323 |
| | 32 | 1073.535 | 2005.292 | 1349.003 |
| Average $T_{pro}/T_{anc}$ | | 1 | 1.74 | 1.05 |

matrix. For other modes of Intra4 × 4, 7-tap filter is employed, thus $7 \times 4 \times 4$ elements are needed to define the filtering coefficient matrix for each mode. Similarly, for Intra8 × 8, 13-tap filter is used, and the filtering coefficient matrix has $13 \times 8 \times 8$ elements for each mode. The bit depth of the element is 2 bytes. Therefore, to store these filtering coefficient matrices, we need $2 \times (2 \times (9 \times 4 \times 4) + 7 \times (7 \times 4 \times 4) + 9 \times (13 \times 8 \times 8)) = 17{,}120$ bytes.

To evaluate the computational complexity, the encoding times for several sequences, including the original KTA platform and our proposed platform, are tabulated in Table 4. Without any optimization to the source code, the average encoding time of the proposed scheme is near 1.7 times larger than the one in H.264/AVC. For an improved version with skipping of some unnecessary rate-distortion process, the average ratio of the encoding time is 1.05 over KTA while the same coding results are kept.

## 5. Conclusion and future work

In this paper, we propose a novel intra prediction algorithm via position-dependent filtering to better remove the spatial redundancies. Different from the existing schemes, in our proposed PDF, different sets of filtering coefficients are employed for samples with different coordinates related to one coding sub-block. For each intra prediction mode, the set of linear filtering coefficients for each position is obtained from off-line training using the least square method. Moreover, to further reduce the algorithm complexity, a simplified PDF is proposed, which reduces the filter taps by discarding some reference samples with minor contribution to intra prediction. Experimental results demonstrate that both PDF and sPDF exhibit much better decorrelation ability and achieves a great coding gain while the computational complexity increase can be acceptable.

In the future work, further improvements of our proposed PDF will be investigated regarding to the following two issues: (1) more efficient method for calculating the filtering coefficients, and (2) applicable adaptively updated method of the filtering coefficients to adapt to the local video content characteristics.

## Acknowledgments

The work was supported in part by National Natural Science Foundation Research Program of China under Grant (Nos. 60833013, 60803068) and National Basic Research Program of China (973 Program, 2009CB320903). The authors would like to thank Dr. Shawmin Lei for providing some good suggestions.

## References

[1] A. Netravali, J. Limb, Picture coding: a review, Proc. IEEE 68 (3) (1980) 366–406.
[2] ITU-T. Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video. ITU-T Recommendation H.262|ISO/IEC 13818-2 (MPEG-2), 1995.
[3] G. Bjøntegaard, Coding Improvement by Using 4 × 4 Blocks for Motion Vectors and Transform, ITU-T Q.15/SG16 VCEG, Doc. Q15-C23, December 1997.
[4] D. Wang, Improvement of H.26L Intra Block Prediction, ITU-T SG16/Q.6 VCEG, VCEG-L09, January 2001.
[5] G. Conklin, New Intra Prediction Modes, ITU-T Q.6/SG16 VCEG, VCEG-N54, September 2001.
[6] T. Wiegand, G. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H.264/AVC video coding standard, IEEE Trans. Circuits Syst. Video Technol. 13 (7) (2003) 560–576.
[7] K. Ugur, K. Andersson, A. Fuldseth, Description of Video Coding Technology Proposal by Tandberg, Nokia, Ericsson, ITU-T Q.6/SG16, ISO/IEC JTC1/SC29, Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-A119, April 2010.
[8] K. McCann, W. Han, I. Kim, Samsung's Response to the Call for Proposals on Video Compression Technology, ITU-T Q.6/SG16, ISO/IEC JTC1/SC29, Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-A124, April 2010.
[9] L. Liu, Multiple Predictor Sets for Intra Coding, ITU-T Q.6/SG16, ISO/IEC JTC1/SC29, Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-A022, April 2010.

[10] S. Taichiro, A. Tanizawa, T. Chujoh, T. Yamakage, Improvement of Bidirectional Intra Prediction, ITU-T Q.6/SG16 VCEG, VCEG-AG08, October 2007.
[11] Y. Zhang, L. Zhang, S. Ma, D. Zhao, W. Gao, Context-adaptive Pixel Based Prediction for Intra-frame Encoding, in: ICASSP, March 2010, pp. 898–901.
[12] L. Liu, Y. Liu, E. Delp, Enhanced intra prediction using context-adaptive linear prediction, in: PCS, November 2007.
[13] S. Yu, Y. Gao, J. Chen, J. Zhou, Distance-based weighted prediction for H.264 intra coding, in: ICALIP, July 2008, pp. 1477–1480.
[14] L. Wang, L. Po, Y. Uddin, K. Wong, S. Li, A novel weighted cross prediction for H.264 intra coding, in: ICME, May 2009, pp. 165–168.
[15] S. Yu, C. Chrysafis, New Intra Prediction Using Intra-Macroblock Motion Compensation, Tech. Rep., JVT-C151, May 2002.
[16] T. Tan, C. Boon, Y. Suzuki, Intra Prediction by Template Matching, in: ICIP, October 2006, pp. 1693–1696.
[17] L. Wei, M. Levoy, Fast texture synthesis using tree-structured vector quantization, in: SIGGRAPH, July 2000, pp. 479–488.
[18] T. Tan, C. Boon, Y. Suzuki, Intra prediction by averaged template matching predictors, in: CCNC, IEEE, January 2007, pp. 405–409.
[19] Y. Guo, Y. Wang, Q. Li, Priority-based template matching intra prediction, in: ICME, June 2008, pp. 1117–1120.
[20] Y. Zheng, P. Yin, O. Escoda, X. Li, C. Gomila, "Intra prediction using template matching with adaptive illumination compensation, in: ICIP, 2008, pp. 125–128.
[21] J. O'Neal, T. Natarajan, Coding isotropic images, IEEE Trans. Inform. Theory IT-23 (1977) 697–707.
[22] Y. Ye, M. Karczewicz, Improved H.264 intra coding based on bi-directional intra prediction, directional transform, and adaptive coefficient scanning, in: ICIP, October 2008, pp. 2116–2119.
[23] jm11.0kta2.6, October 2009. <http://www.iphome.hhi.de/suehring/tml/download/KTA>.
[24] G. Bjontegaard, Calculation of Average PSNR Differences between RD-Curves, ITU-T Q.6/SG16 VCEG, VCEG-M33, April 2001.