

ON THE ANALYSIS AND DESIGN OF MOTION SAMPLING STRUCTURE FOR ADVANCED MOTION-COMPENSATED PREDICTION

Yu-Chen Tseng, Chung-Hao Wu, Yi-Wen Chen, Tse-Wei Wang, and Wen-Hsiao Peng

Department of Computer Science, National Chiao Tung University, Taiwan

ABSTRACT

This paper addresses the problem of improving motion sampling efficiency for motion-compensated prediction (MCP). We provide a theoretical framework for analyzing the effect of motion sampling structure on MCP efficiency. It is shown that the sampling grid induced by the quadtree partition in H.264/AVC is suboptimal. To improve sampling efficiency, we propose a new pattern, which provides sampling points at both the center and the top-left corner of a macroblock. When contrasted with conventional $N \times N/2$ block partition, the proposed scheme performs consistently and significantly better in subjective and objective quality.

Index Terms— Motion Sampling, H.264/AVC, OBMC

1. INTRODUCTION

Motion-compensated prediction (MCP) is widely used in video compression for removing temporal redundancy. Its process can be viewed as consisting of sparse motion sampling followed by the reconstruction of a temporal predictor. The latter is usually accomplished by block-based motion compensation (BMC), which implements nearest-neighbor motion interpolation. Because this crude interpolator is far from being ideal, it often introduces motion uncertainty.

A number of advanced MCP schemes have thus been proposed. Sullivan *et al.* [10] introduced a control grid interpolation (CGI) that bilinearly interpolates between the known motion vectors (MVs) of control points. Nogaki *et al.* [8] and Orchard *et al.* [9], on the other hand, computed an optimal linear estimate of the prediction pixel using the pixel values derived from its neighboring MVs. This approach is known as overlapped block motion compensation (OBMC). In addition, there are hybrid schemes [2][3][4][6][7], which combine BMC, CGI or OBMC to address the non-stationarity of the motion field.

Although the reconstruction process was well studied, the efficiency of motion sampling has rarely been addressed. This is due in part that most MCP schemes require motion samples to be taken on a rectangular grid. For example, difficulties arise if CGI needs to map between two arbitrary quadrangles, or if OBMC (or BMC) has to rely on MVs that may be arbitrarily positioned. Faced with the inability to freely move grid points, some [4][5] resort to hierarchical quadtree segmentation, while others [2][3] incorporate interpolated MVs where higher sampling density is in need.

To overcome this problem, we devised in [1] a parametric OBMC scheme, which enables MVs sampled on an irregular grid to be utilized for MCP without constraint. In doing so, it is found that the sampling grid induced by the quadtree partition in H.264/AVC is suboptimal: it tends to be denser around macroblock centers and sparser in the other areas. In the first part of this paper, we propose

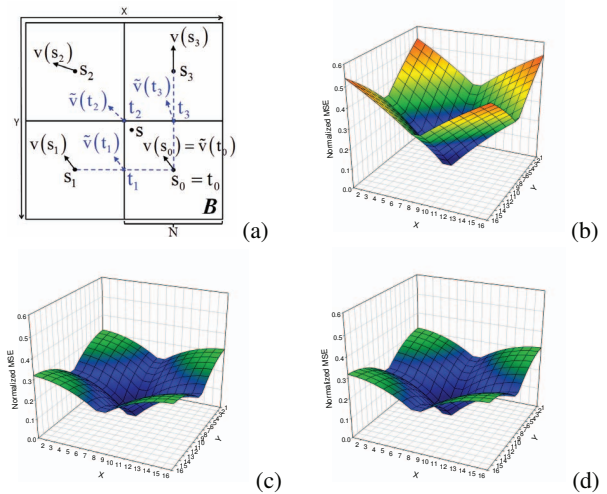


Fig. 1. Mean-square prediction error surfaces of block B produced with (b) BMC (c) CGI/OBMC and (d) IG-OBMC. The block size used for motion compensation is 16×16 .

a theoretical framework to analyze the effect of motion sampling structure on MCP efficiency. We then apply some of these results to design a new sampling pattern, which provides sampling points at both the center and the top-left corner of a macroblock, and when applied repeatedly, forms a rhombic lattice. When contrasted with $N \times N/2$ block partition, the proposed scheme performs consistently and significantly better in subjective and objective quality.

This paper is organized as follows: Section 2 analyzes the efficiency of various MCP schemes from a theoretical perspective. Section 3 points out the problem of the quadtree partition in H.264/AVC and introduces a new sampling pattern derived from rhombic lattice. We then assess the subjective and objective quality of the proposed scheme in Section 4 before concluding this paper with a summary of our work.

2. MOTION-COMPENSATED PREDICTION

In this section, we use an estimation-theoretical approach to analyze how motion sampling structure may affect the efficiency of MCP.

2.1. Signal Model

To facilitate the analysis, we will adopt the signal model proposed in [11], which assumes that the *autocorrelation function* of the intensity and motion fields takes the following quadratic and exponential

forms, respectively:

$$E[I_k(\mathbf{s}_1)I_k(\mathbf{s}_2)] = \sigma_I^2 \left(1 - \frac{\|\mathbf{s}_1 - \mathbf{s}_2\|^2}{K} \right) \quad (1a)$$

$$E[v_x(\mathbf{s}_1)v_x(\mathbf{s}_2)] = E[v_y(\mathbf{s}_1)v_y(\mathbf{s}_2)] = \sigma_m^2 \rho_m^{\|\mathbf{s}_1 - \mathbf{s}_2\|_1} \quad (1b)$$

where $I_k(\mathbf{s})$ represents the intensity value of pixel $\mathbf{s} = (x(\mathbf{s}), y(\mathbf{s}))^T$ in frame k ; $\mathbf{v}(\mathbf{s}) = (v_x(\mathbf{s}), v_y(\mathbf{s}))^T$ denotes its motion vector; and $\{\sigma_I^2, K\}$ and $\{\sigma_m^2, \rho_m\}$ are their respective variance and correlation coefficient. In the temporal dimension, we further assume $I_k(\mathbf{s}) = I_{k-1}(\mathbf{s} + \mathbf{v}(\mathbf{s}))$. Moreover, block motion vectors are approximated as the motion at block centers, and in that regard, block-based motion estimation is seen as a motion sampler.

2.2. Mean-Square Prediction Error

Given these assumptions, we next examine the prediction error for BMC, CGI [10], OBMC [9] and IG-OBMC [3]. Assume at first the sampling structure is a square lattice. Such is the case when an image is divided into equally spaced square blocks for motion estimation. Then, with reference to Fig. 1 (a), the prediction error of pixel \mathbf{s} , $\mathbf{s} \in \mathcal{B}$ for the four schemes can be expressed respectively as

$$d^{BMC}(\mathbf{s}) = I_k(\mathbf{s}) - I_{k-1}(\mathbf{s} + \mathbf{v}(\mathbf{s}_0))$$

$$d^{CGI}(\mathbf{s}) = I_k(\mathbf{s}) - I_{k-1} \left(\mathbf{s} + \sum_{i=0}^3 w_i^{(c)}(\mathbf{s}) \mathbf{v}(\mathbf{s}_i) \right)$$

$$d^{OBMC}(\mathbf{s}) = I_k(\mathbf{s}) - \sum_{i=0}^3 w_i^{(o)}(\mathbf{s}) I_{k-1}(\mathbf{s} + \mathbf{v}(\mathbf{s}_i))$$

$$d^{IG-O}(\mathbf{s}) = I_k(\mathbf{s}) - \sum_{i=0}^3 w_i^{(ig)}(\mathbf{s}) I_{k-1}(\mathbf{s} + \tilde{\mathbf{v}}(\mathbf{t}_i))$$

where $\{w_i^{(c)}(\mathbf{s})\}$ are chosen such that $\sum w_i^{(c)}(\mathbf{s}) \mathbf{v}(\mathbf{s}_i)$ forms a vector LMMSE estimate of $\mathbf{v}(\mathbf{s})$ subject to the unit gain constraint¹. By a similar approach, the weighting coefficients $\{w_i^{(o)}(\mathbf{s})\}$ and $\{w_i^{(ig)}(\mathbf{s})\}$ are derived to linearly estimate $I_k(\mathbf{s})$ based on the data sets $\{I_{k-1}(\mathbf{s} + \mathbf{v}(\mathbf{s}_i))\}$ and $\{I_{k-1}(\mathbf{s} + \tilde{\mathbf{v}}(\mathbf{t}_i))\}$, respectively. Particularly, in computing $\{w_i^{(ig)}(\mathbf{s})\}$ the motion vectors at \mathbf{t}_i , $i = 1, 2, 3$ are taken to be known, while during actual motion compensation they are interpolated from those of nearby block centers (with the results denoted by $\tilde{\mathbf{v}}(\mathbf{t}_i)$).

The mean-square prediction error (MSE) for the four MCP schemes can be evaluated by using (1), although the algebra is a bit tedious. We shall thus use CGI as an example to indicate the main idea without going into formal details. To start off, the vector LMMSE estimator for $\mathbf{v}(\mathbf{s})$ is firstly found by combining the scalar estimator for each of its components. As such, $w_i^{(c)}(\mathbf{s})$ is a matrix-valued function (of dimension 2×2). However, a great simplification can be made since (a) the horizontal and vertical motion fields are independent of each other and (b) they share an identical signal model as hinted in (1b). The former makes the matrix become diagonal while the latter further equalizes the diagonal elements. Together the two conditions reduce $w_i^{(c)}(\mathbf{s})$ to a scalar, with its value given by the i th element of

$$\mathbf{w}^{(c)}(\mathbf{s}) = \mathbf{R}^{-1} \left[\mathbf{P} - \mathbf{U} \left(\frac{\mathbf{U}^T \mathbf{R}^{-1} \mathbf{P} - 1}{\mathbf{U}^T \mathbf{R}^{-1} \mathbf{U}} \right) \right], \quad (2)$$

¹We consider this Wiener filter rather than bilinear filter [10] since our interest is in determining the theoretic limit of CGI.

where \mathbf{U} is a unit vector and $\mathbf{R}_{ij} = E[v_x(\mathbf{s}_i)v_x(\mathbf{s}_j)]$ and $\mathbf{P}_j = E[v_x(\mathbf{s})v_x(\mathbf{s}_j)]$ for $0 \leq i, j \leq 3$.

To complete the evaluation of $E[\|d^{CGI}(\mathbf{s})\|^2]$, we still need to know $E[I_k^2(\mathbf{s})]$, $E[I_{k-1}^2(\mathbf{s} + \sum w_i^{(c)}(\mathbf{s}) \mathbf{v}(\mathbf{s}_i))]$, and $E[I_k(\mathbf{s})I_{k-1}(\mathbf{s} + \sum w_i^{(c)}(\mathbf{s}) \mathbf{v}(\mathbf{s}_i))]$. The first two terms, according to (1a), are simply σ_I^2 , while the last one can be computed by substituting (1) and (2) into (3).

$$E \left[I_{k-1}(\mathbf{s} + \mathbf{v}(\mathbf{s})) I_{k-1} \left(\mathbf{s} + \sum_{i=0}^3 w_i^{(c)}(\mathbf{s}) \mathbf{v}(\mathbf{s}_i) \right) \right] \quad (3)$$

$$= \sigma_I^2 E \left[1 - 2K^{-1} \left(\sum_{i=0}^3 w_i^{(c)}(\mathbf{s}) (v_x(\mathbf{s}) - v_x(\mathbf{s}_i)) \right)^2 \right]$$

where we have used the fact that $\sum w_i^{(c)}(\mathbf{s}) = 1$. A straightforward computation then gives

$$E \left[\|d^{CGI}(\mathbf{s})\|^2 \right] = \frac{f}{2} \sum_{0 \leq i, j \leq 3} w_i^{(c)}(\mathbf{s}) w_j^{(c)}(\mathbf{s}) \left(1 - \rho_m^{\|\mathbf{s} - \mathbf{s}_i\|_1} - \rho_m^{\|\mathbf{s} - \mathbf{s}_j\|_1} + \rho_m^{\|\mathbf{s}_i - \mathbf{s}_j\|_1} \right), \quad (4)$$

with the scaling factor $f = 8\sigma_I^2 \sigma_m^2 K^{-1}$. Following similar derivations to those for CGI, we can calculate the MSE for the other schemes as

$$E \left[\|d^{BMC}(\mathbf{s})\|^2 \right] = f \left(1 - \rho_m^{\|\mathbf{s} - \mathbf{s}_0\|_1} \right)$$

$$E \left[\|d^{OBMC}(\mathbf{s})\|^2 \right] = E \left[\|d^{CGI}(\mathbf{s})\|^2 \right]_{\mathbf{w}^{(c)}(\mathbf{s}) = \mathbf{w}^{(o)}(\mathbf{s})}$$

$$E \left[\|d^{IG-O}(\mathbf{s})\|^2 \right] = \sum_{0 \leq i, j \leq 3} w_i^{(ig)}(\mathbf{s}) w_j^{(ig)}(\mathbf{s}) E[I_{\Delta}(i)I_{\Delta}(j)]$$

where $I_{\Delta}(i) = I_k(\mathbf{s}) - I_{k-1}(\mathbf{s} + \tilde{\mathbf{v}}(\mathbf{t}_i))$ and $E[I_{\Delta}(i)I_{\Delta}(j)]$ can be expanded and evaluated term by term through a calculation similar to (3).

It is interesting that the MSE of OBMC exhibits the same form as that of CGI, with $\mathbf{w}^{(o)}(\mathbf{s})$ substituting for $\mathbf{w}^{(c)}(\mathbf{s})$. Somewhat surprisingly, $\mathbf{w}^{(o)}(\mathbf{s})$ is found to be equal to $\mathbf{w}^{(c)}(\mathbf{s})$, suggesting that OBMC and CGI have identical prediction efficiency and that the OBMC filter $\mathbf{w}^{(o)}(\mathbf{s})$ is also a good motion interpolator. Nevertheless, OBMC is generally preferable to CGI. The reasons are twofold. First, the true motion for every pixel is not easily accessible, which makes it difficult to estimate $\mathbf{w}^{(c)}(\mathbf{s})$ for CGI. Second, OBMC can not only alleviate motion uncertainty, but it also serves to attenuate quantization noises in reference pictures. These arguments also explain why OBMC normally outperforms CGI in practice.

2.3. Assessing Prediction Efficiency

Fig. 1 (b)-(d) depict the MSE surfaces over block \mathcal{B} . Comparison with BMC shows that OBMC and CGI can indeed help to equalize prediction errors across the block and provide a substantial reduction in MSE. However, they cannot work miracles: the error variance still increases towards the outer region of the prediction block. Although IG-OBMC attempts to solve this problem by incorporating interpolated motion vectors for OBMC, the improvement is quite limited, as is evident from its MSE surface. In fact, its MSE may even be increased, as will be shown next.

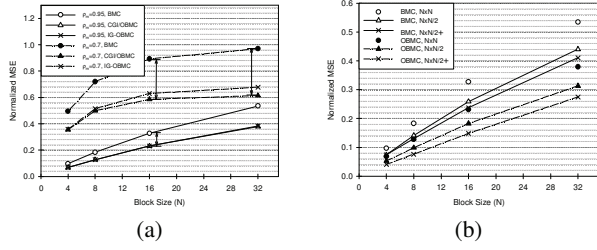


Fig. 2. MSE Comparison: (a) MSE as a function of block size for different ρ_m values, and (b) MSE for different sampling patterns and MCP methods.

In Fig. 2 (a), the overall MSE is plotted as a function of block size for different ρ_m values, in order to see how the change in (motion) sampling density and motion statistics may affect prediction efficiency. The y-axis has been normalized since all the MSE models include the scaling factor f . This also indicates the obvious relation between the MSE, σ_l^2 , σ_m^2 and K . As expected, the gain of OBMC/CGI/IG-OBMC over BMC increases drastically as motion uncertainty increases with increasing block size or motion randomness². It is also seen that IG-OBMC performs very close to OBMC/CGI. Its MSE is slightly higher due mainly to the use of suboptimal bilinear filter for motion interpolation. Nonetheless, this result has an interesting interpretation—that is, the predictor quality can hardly be improved without introducing "new" motion information. Recall that $\{\tilde{v}(t_i)\}$ are interpolated from the same set of motion vectors $\{v(s_i)\}$ used for OBMC/CGI.

3. IMPROVING MOTION SAMPLING

A question that naturally arises, when additional motion vectors are allowed for prediction, is how to efficiently sample the motion field to attain better prediction (in terms of MSE). The simplest yet most widely used approach is the quadtree block partition, such as that adopted by H.264/AVC. However, one main problem with this scheme is that it is specifically tailored to rectangular BMC, which somewhat restricts the permissible motion sampling structures and hence limits the best achievable performance. This can be better explained using the example shown in Fig. 3, where the motion sampling patterns for the 16x16 and 16x8 modes and their respective error surfaces (with OBMC) are contrasted.

Apparently, the 16x8 pattern is suboptimal in that the extra motion vector tends to minimize the already reduced prediction errors around block center (compare Fig. 3 (d) and (e)). Although the net result is a further reduction in MSE, the gain has not reached its maximum. In fact, the motion vector can be put to best use if the sampling grid is redistributed as a rhombic lattice (see Fig. 3 (c)), a result that is intuitively obvious and can be verified numerically by the procedures in Section 2. Essentially, the idea is to introduce "new" motion vectors where they are needed most. Recall that motion uncertainty is the highest at block corners.

Fig. 2 (b) compares the theoretical performance for different sampling patterns, where the new one is denoted as $N \times N/2^+$ with N indicating the block size. The naming suggests that it has the same number of motion vectors per block as the $N \times N/2$ mode. However, in applying BMC, its basic unit is a rhombus rather than a rectangle.

²The motion randomness increases with decreasing ρ_m .

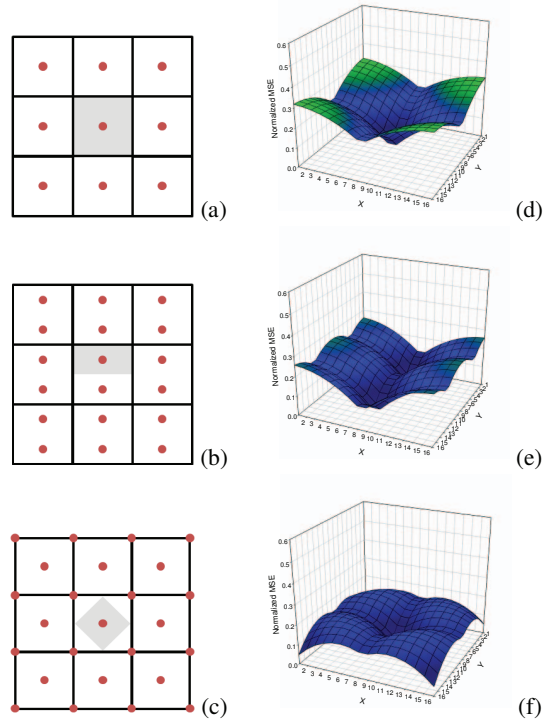


Fig. 3. Motion sampling patterns for the (a)16x16, (b)16x8, and (c)16x8⁺ modes and their theoretical MSE surfaces associated with OBMC.

From the results, we see that the $N \times N/2^+$ pattern is indeed more efficient. The MSE drops by 6-8% as compared to that with the $N \times N/2$ mode. Moreover, it can be used to good advantage especially with OBMC or CGI. Not only is the MSE improvement more significant (12-18%), but also the subjective quality is expected to be improved further since its error surface tends to be nearly flat as shown in Fig. 3 (f).

Up to now, we have considered only the case when a fixed sampling pattern is repeatedly applied to every macroblock. In practice, sampling pattern is switchable at macroblock level and the resulting grid may be irregular. In this case it is unknown how much gain the $N \times N/2^+$ pattern can actually offer. For a deeper analysis, we include it as an alternative option to quadtree patterns and employ a greedy algorithm for mode decision. The notion is to increase sampling density incrementally by a rate-distortion optimization approach. As a first step, each macroblock in the prediction frame \mathcal{F} is assigned with one MV. Then, the algorithm decides based on a rate-distortion criterion³ which macroblock b , $b \in \mathcal{F}$ deserves more vectors; the selected block is updated with the pattern exhibiting the highest rate-distortion gain. The process is repeated recursively until a target number of MVs N is achieved. The algorithm is greedy because at every iteration, it chooses the candidate block without considering the consequence to the following iterations. It is also suboptimal in that the evolving of sampling pattern for each block is subject to the transition rule depicted in Fig. 4. Nevertheless, it is good enough to serve our evaluation purpose.

³We define the criterion as the ratio of MSE improvement to the extra number of MVs incurred due to the increase of sampling density.

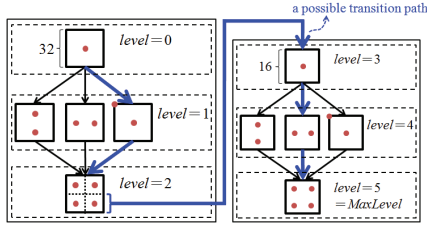


Fig. 4. Transition rule for incremental motion refinement.

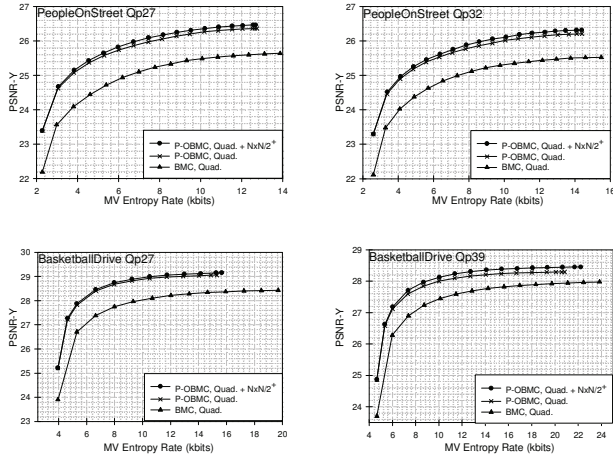


Fig. 5. Predictor PSNR at various QP settings and MV entropy rates.

4. RESULTS AND CONCLUDING REMARKS

In this section, we test the $N \times N/2^+$ pattern against the quadtree partition in H.264/AVC for a number of standard HD sequences. Parametric OBMC (P-OBMC) [1] is adopted as the MCP method, with BMC's performance used as a baseline. Motion estimation for the grid points at levels 0-2 and levels 3-5 has a block size of 16×16 and 8×8 , respectively, and motion search accuracy is of quarter-pel. To simulate quantization effects, the reference frame is coded by H.264/AVC.

In Fig. 5, the predictor quality is shown as a function of the MV entropy rate. The objective results indicate that our scheme, when applied jointly with the quadtree patterns, performs consistently better although the gain is only about 0.2-0.3dB. This is essentially a consequence of the hybridization. To substantiate our claim, Table 1 provides the results with the single use of the $N \times N/2^+$ or $N \times N/2$ pattern ($N=16$). It is seen that there is a 0.2-2.2dB PSNR improvement (or 5-40% MSE reduction) across different QP settings. The gain is most obvious for sequences having complex motion, due largely to the effective removal of motion uncertainty. Moreover, the empirical data also agree well with our theoretical estimates. Fig. 5 further shows close-ups of the predictors generated by these two patterns. As expected, the visual quality of the $N \times N/2^+$ improves significantly.

Summarizing, in this paper, we have analyzed the effect of motion sampling structure on MCP efficiency. The sampling grid induced by the quadtree partition in H.264/AVC is shown to be suboptimal. A new sampling pattern, which we call $N \times N/2^+$, has thus been proposed to improve motion sampling efficiency. When contrasted

Table 1. PSNR Gain and MSE Reduction Rate

QP	22		27		32		37	
	PSNR/MSE	Δ dB	Δ dB	%	Δ dB	%	Δ dB	%
Bas.Drive	2.2	40	2.1	39	2.0	37	1.8	34
People.St.	0.2	5	0.3	7	0.4	9	0.5	11
Cactus	0.9	19	0.9	19	0.9	18	0.7	15
Traffic	0.9	19	0.9	19	0.8	18	0.7	16



Fig. 6. Perceptual quality of Frame 14 of the People on Street with the (a) $N \times N/2$ and (b) $N \times N/2^+$ patterns.

with conventional $N \times N/2$ block partition, it performs consistently and significantly better in subjective and objective quality.

5. REFERENCES

- [1] Y. W. Chen, T. W. Wang, Y. C. Tseng, W. H. Peng, and S. Y. Lee, "A Parametric Window Design for OBMC with Variable Block Size Motion Estimates," *Proc. Int. Workshop Multimedia Signal Processing*, 2009.
- [2] C.-C. Cheng, W.-L. Hwang, Z. Shen, and T. Xia, "Subjective and Objective Comparison of Advanced Motion Compensation Methods for Blocking Artifact Reduction in a 3-D Wavelet Coding System," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 9, pp. 1134-1141, 2006.
- [3] B.-D. Choi, J.-W. Han, and S.-J. Ko, "Irregular-Grid-Overlapped Block Motion Compensation and Its Practical Application," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 8, pp. 1221-1226, 2009.
- [4] G. Heising, D. Marpe, H. L. Cycon, and A. P. Petukhov, "Wavelet-Based Very Low Bit Rate Video Coding Using Image Warping and Overlapped Block Motion Compensation," *Proc. IEE Vis., Image Signal Process.*, vol. 148, no. 2, pp. 93-101, 2001.
- [5] C.-L. Huang and C.-Y. Hsu, "A New Motion Compensation Method for Image Sequence Coding Using Hierarchical Grid Interpolation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 1, pp. 42-52, 1994.
- [6] P. Ishwar and P. Moulin, "Switched Control Grid Interpolation for Motion Compensated Video Coding," *Proc. Int. Conf. Image Processing*, pp. 650-653, 1997.
- [7] T.-Y. Kuo and C.-C. J. Kuo, "Fast Overlapped Block Motion Compensation with Checkerboard Block Partitioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 6, pp. 705-712, 1998.
- [8] S. Nogaki and M. Ohta, "An Overlapped Block Motion Compensation for High Quality Motion Picture Coding," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 184-187, 1992.
- [9] M. T. Orchard and G. J. Sullivan, "Overlapped Block Motion Compensation: An Estimation-Theoretic Approach," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 693-699, 1994.
- [10] G. J. Sullivan and R. L. Baker, "Motion Compensation for Video Compression Using Control Grid Interpolation," *Proc. ICASSP*, pp. 2713-2716, 1991.
- [11] B. Tao and M. T. Orchard, "A Parametric Solution for Optimal Overlapped Block Motion Compensation," *IEEE Trans. Image Processing*, vol. 10, pp. 341-350, 2001.