

STATISTICAL LEARNING BASED INTRA PREDICTION IN H.264

Cheolhong An and Truong Q. Nguyen

UCSD, ECE Dept, La Jolla, California 92093-0407

chan@ucsd.edu, nguyent@ece.ucsd.edu

<http://videoprocessing.ucsd.edu/>

ABSTRACT

In this paper, we improve the performance of intra prediction and simplify mode decision procedure at the same time. For these works, we apply a statistical learning method such as Support Vector Machines for Regression (SVR) to improve the performance of current H.264 intra prediction via batch learning. In addition, we only use single Macro Block type and one intra prediction mode with high prediction performance to simplify mode decision procedure. In our knowledge, this work is the first approach to apply a statistical learning method for prediction of video sequences. Therefore, we introduce theoretical backgrounds of SVR, and show the possibility of this challenge for video compression. From the experimental results, statistical learning based intra prediction improves significantly the average Peak Signal-to-Noise Ratio of intra prediction than the performance of current H.264.

Index Terms— H.264, Intra prediction, Statistical learning, Support vector machines.

1. INTRODUCTION

Many works related with intra prediction mainly propose methods to improve the performance of current H.264 intra prediction [1] or to decide the best intra prediction mode with low complexity and minimum loss of performance. The methods in [2][3] propose fast mode decision in frequency domain and these in [4] [5] represent how pixel domain intra predictions are correspond to DCT domain operation. Especially, reference [4] proposes additional prediction modes with increased mode decision complexity. Authors of [6] match a template for 2x2 block with similarity measure in pixel domain. The approaches in [7][8] imitate motion estimation and compensation to search the best matching block among the neighbor blocks with sub-pel accuracy to improve intra prediction performance. Reference [9] proposes a hybrid method to use both pixel based intra prediction of current H.264 and block matching.

In this paper, we improve the performance of intra prediction and simplify mode decision procedure at the same time. We apply a statistical learning method such as Support Vector Machines for Regression (SVR) [10, 11, 12] to improve the

performance of current H.264 intra prediction via batch learning. Support vector machines for classification and regression have been developed under profound theoretical background and they are successfully applied to many classification and time series prediction [12]. In order to simplify mode decision procedure, we only use single Macro Block (MB) type and one prediction mode, that is, SVR with high prediction performance. We note that average PSNR of intra prediction is around 26dB at the lowest Quantization Parameter (QP) which is not higher than we expect. Therefore, previous works [6, 7, 8, 9] improve the performance in PSNR sense but the improvement is under 1dB. In this proposed method, we significantly improve the performance more than 1dB.

The rest of this paper is organized as follows. We briefly introduce current intra prediction of H.264 in section 2. In section 3, SVR is considered as a batch learning method and SVR is trained in DCT domain and then applied in intra prediction. Experimental results of SVR based prediction are presented in section 4. Section 5 concludes the paper.

2. INTRA PREDICTION OF H.264

H.264 [1] uses 9 directional intra prediction modes for 4x4 block and 4 intra prediction modes for 16x16 MB. Key idea of the H.264 intra prediction is extrapolation of the pixels which are on row and column directly adjacent to the current block. All the intra prediction modes of each MB type are performed in pixel domain through directional extrapolation. Usually, the best prediction mode is decided via Rate-Distortion (R-D) optimization to minimize the Lagrangian cost [13][14]. The prediction errors are transformed by Discrete Cosine Transform (DCT) and then the DCT coefficients are quantized. The quantization indexes are coded by entropy coding such as Universal Variable Length Coding (UVLC) and Context Adaptive Binary Arithmetic Coding (CABAC) [1][15]. Reconstructed pixels are obtained from adding predicted pixels and reconstructed errors which result from decoding, inverse quantization and inverse DCT transformation. Note that these reconstructed pixels are used for intra prediction instead of original pixels in order to prevent a drift problem between encoder and decoder and deblocking filter is not applied to the reconstructed pixels. This note is also applied to intra prediction of SVR.

This work is supported by CWC and matching fund from UC Discovery program

3. INTRA PREDICTION VIA STATISTICAL LEARNING METHOD

3.1. Support Vector Machine

We consider SVR as a batch learning method. SVMs are statistical learning tools based on Vapnik-Chervonenkis (VC) theory and Structural Risk Minimization (SRM) principles [12]. SRM is an inductive principle for model selection which is used for learning from finite training data sets. It describes a general model of capacity control and provides a trade-off between hypothesis space complexity (VC dimension of approximating functions) and the quality of fitting the training data. Statistical learning theory and SVMs show that the regularization networks also can approximately implement SRM principles when an optimal regularization parameter has been chosen [16]. Therefore, SVR solves a Regularized Risk (summation of empirical risk and regularizer) Minimization (RRM) problem to estimate linear function $f(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b$ for ϵ -incentive loss function as follows [12]:

$$\begin{aligned} \min_{\mathbf{w}, b, \xi, \xi^*} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N (\xi_i + \xi_i^*) \\ \text{s.t.} \quad & \langle \mathbf{w}, \Phi(\mathbf{x}_i) \rangle - y_i \leq \epsilon + \xi_i \\ & y_i - \langle \mathbf{w}, \Phi(\mathbf{x}_i) \rangle \leq \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \geq 0, i = 1, \dots, N \end{aligned} \quad (1)$$

where C is inverse regularization parameter and Φ is a nonlinear mapping function from input data \mathbf{x} into a high-dimensional feature space. N denotes the number of training samples and $\xi_i^{(*)}$ are slack variables to allow violation of condition which is called soft margin. y_i are output corresponding to input data \mathbf{x}_i and ϵ is a parameter which denotes zero loss if absolute value of prediction error $|y_i - f(\mathbf{x}_i)|$ is smaller than ϵ . The optimization problem (1) is a quadratic convex optimization problem and its solutions are global optimal solutions which is main feature of SVMs. The primal optimization problem (1) can be solved as a primal optimization view [17] or dual optimization view through the Lagrangian duality [12]. Two optimization views derive the same regression function $f(\mathbf{x})$ as a solution of (1) as follows:

$$\begin{aligned} f(\mathbf{x}) &= \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b = \sum_{i=1}^N (\alpha_i^* - \alpha_i) \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}) \rangle + b \\ &= \sum_{i \in SV}^{\#SV} (\alpha_i^* - \alpha_i) \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}) \rangle + b \\ &= \sum_{i \in SV}^{\#SV} (\alpha_i^* - \alpha_i) k(\mathbf{x}_i, \mathbf{x}) + b \end{aligned} \quad (2)$$

where $\mathbf{w} = \sum_{i=1}^N (\alpha_i^* - \alpha_i) \Phi(\mathbf{x}_i)$ and $\alpha_i^{(*)}$ are dual optimal solutions of a dual optimization problem. Note that input data \mathbf{x}_i which have non-zero $\alpha_i^{(*)}$ are called as Support

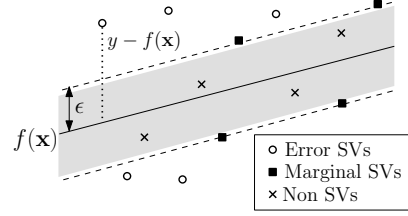


Fig. 1. Classifications of training input data \mathbf{x}_i .

Vectors (SVs). After solving (1), training data \mathbf{x}_i are classified into three types according to the absolute prediction error $|y_i - f(\mathbf{x}_i)|$: marginal SVs, error SVs and non SVs which are illustrated in Figure 1. Thus, if the absolute prediction error is equal to ϵ , the input data \mathbf{x}_i are called as marginal SVs and if it is larger than ϵ , the input data are error SVs and otherwise, the input data are not SVs whose $\alpha_i^{(*)}$ are zero. Consequently, only support vectors among the training data contribute regressor output as (2) which gives sparse solutions to SVMs having the ϵ -incentive loss function. However, the sparsity is only achieved by specific loss functions which have zero-gradient loss functions. Reference [17] denotes that dual optimal solutions $\alpha_i^{(*)}$ are related with gradient of a loss function, that is, $\alpha_i^{(*)}$ are zeros if gradients of a loss function at \mathbf{x}_i are zeros. In this paper, we only consider the ϵ -incentive loss function for sparse solutions. In order to reduce complexity of inner products in the high dimensional feature space in (2), the kernel trick [12] is introduced to compute the inner products in the feature space through a kernel function on input data \mathbf{x}_i as follows in (3): $k(\mathbf{x}_i, \mathbf{x}) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}) \rangle$. Here, Radial Basis Function (RBF) is considered as a kernel function in this paper: $k(\mathbf{x}_i, \mathbf{x}) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2}$. Furthermore, a kernel function measures similarity among the SVs \mathbf{x}_i and test data \mathbf{x} .

3.2. Model Selection of Support Vector Machine

Before solving the optimization problem (1), we have to decide the kernel function $k(\cdot)$, kernel parameter γ , loss function parameter ϵ and inverse regularization parameter C which is known as a model selection. In this paper, well-known RBF is used and the other parameters are obtained from Cross Validation (CV). Especially, 5 fold CV is considered which is that training data \mathbf{x}_i are divided into 5 sets and one of 5 sets is used for test and the others are used for training to decide support vectors and their weights $\alpha_i^{(*)}$ for given parameters and this operation is performed 5 times to choose a different test set. Finally, the best parameters which give minimum average Mean Square Error (MSE) through 5 fold CV are applied to the problem (1). Generally, grid-search on the parameters γ , ϵ and C is used for CV [18]. However, references [19, 20, 21] obtained better performance from Genetic Algorithm (GA) for model selections. GA is powerful stochastic search and optimization technique based on the processes of evolution theory. It is excellent for quickly finding an approximate global maximum value. GA uses three

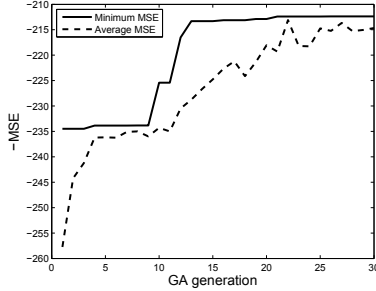


Fig. 2. Example of a model selection in GA.

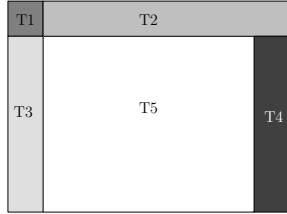


Fig. 3. Classifications of patches in a frame.

operators to generate test patterns: reproduction, crossover and mutation. In this paper, we use GA Matlab tool box which is available in [22]. Since reference [23] denotes that the optimal points of hyper-parameters do not exist uniquely, model parameters are selected to generate fewer SVs and smaller C in the case of the same MSE to find sparser solution and smaller regularized risk in (1). Figure 2 represents that GA generates better test patterns which are closer to the global solutions according to generations because of the smaller (average) minimum MSE. However, it keeps generating new test patterns from three operators to escape local minima.

3.3. Intra Prediction of Support Vector Machine

In this subsection, we decide output and input features of SVR. Output of SVR is a DCT coefficient and input features x_i are DCT coefficients of neighbor MBs. Input features are classified 5 types from T1 to T5 whose classifications are based on available neighbor 8x8 MBs as shown in Figure 3. For example, a left MB is only available in T2 classification. Main structural difference from H.264 [1] is that 8x8 MB type and 8x8 DCT are only applied instead of 16x16 and 4x4 MB types and 4x4 DCT. Furthermore, DCT domain intra prediction is performed with a single prediction method of SVR which is compared with the pixel domain prediction of H.264 with 9 or 4 directional prediction methods. Thus, there are no needs to allocate bits to indicate a MB type and the best intra prediction method in our proposed method. Note that classification types are fixed according to the position of a frame which is already known at decoder.

Figure 4 illustrates that T5 input features and DCT coefficients of a current MB are predicted from incremental intra prediction with inverse zig-zag scan order. The highest DCT

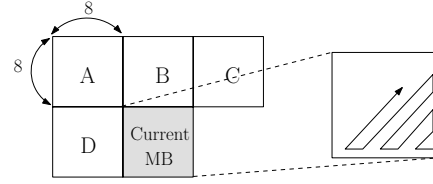


Fig. 4. Patch of SVR intra prediction and incremental intra prediction.

coefficient is only predicted from the neighbor MBs which are denoted as A, B, C and D in Figure 4. DCT coefficients of each MB become a vector via zig-zag scan and then they are concatenated for a input features x_i . If there are no available neighbor MBs, the highest DCT coefficient of T1 classification is coded without intra prediction. Next DCT coefficient with inverse zig-zag scan order is predicted from neighbor MBs and the reconstructed highest DCT coefficient which is obtained from adding a predicted DCT coefficient to the inverse quantized DCT coefficient. Finally, a DC coefficient is predicted from neighbor MBs and AC coefficients of a current MB. Incremental intra prediction with inverse zig-zag scan order overcomes smaller input features. Especially, T1 MB has no neighbor MBs. Therefore, current intra prediction of H.264 [1] subtracts 128 value in pixel domain which is correspond to subtracting a constant value from a DC coefficient in DCT domain. The other directional predictions in pixel domain only subtract some portions of DCT coefficients in DCT domain [5]. Incremental intra prediction overcomes these limitations and utilizes the fact that low frequency DCT coefficients are important. Thus, DC coefficient and low frequency AC coefficients have higher dimension of input features than high frequency AC coefficients.

Figure 5 represents variance of intra prediction errors at zig-zag scan order DCT coefficients of T5 classification in Foreman sequence according to three different prediction methods. In this experiment, 8x8 DCT is only applied for intra prediction errors to all three methods. We use the base-layer of Joint Scalable Video Model (JSVM) [24] for intra prediction of H.264 which is compatible with H.264 [1]. Non-incremental intra prediction of SVR does not utilize current MB information for intra prediction, that is, only uses neighbor MBs from A to D in Figure 4. Incremental intra prediction of SVR has smaller variance of prediction errors in DCT domain than H.264 and non-incremental SVR as shown in Figure 5. Thus, DCT coefficients of a current MB carry very important features to SVR learning system.

4. EXPERIMENTAL RESULTS

In this experiment, we assume that encoder and decoder already have SVs, their weights $\alpha^{(*)}$, b and kernel parameter γ which are needed for prediction as in (3). Foreman sequence is used for training and test. Due to the small number of T1 classification of total frames, every other frames are used for training of T1 classification. We train SVR of T5 and the

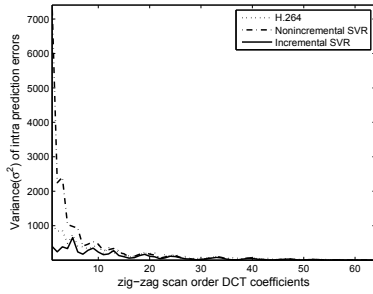


Fig. 5. Variance of intra prediction errors at T5 classification.

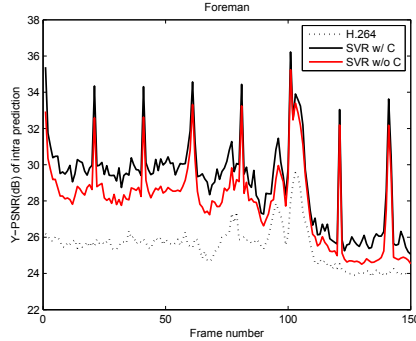


Fig. 6. Y-PSNR(dB) of SVR intra prediction.

other classifications by using every 20th frames and every 5th frames, respectively. RBF kernel is considered for a kernel function and kernel parameter γ , loss function parameter ϵ and regularization parameter C are obtained from 5 fold CV through GA in subsection 3.2. Then, the optimization problem (1) is solved by LIBSVM tools [25] to decide SVs and their weights $\alpha^{(*)}$ for given model parameters.

In Figure 6, the performance of intra prediction of JSVM [24] at the base layer which is compatible with H.264 [1] is compared with the performance of SVR with and without classifications. Here, we only compare PSNR of intra prediction without considering coded bits because side information to indicate the best prediction mode and MB type is not coded in SVR method. Thus, SVR intra prediction only use a 8x8 MB type and one prediction mode. Note that PSNR of intra prediction is obtained from prediction errors E_r at each frame as follows: $10 \log_{10} \frac{255^2}{E_r^2}$. The regular peak in Figure 6 is due to the training of every 20th frames. However, the PSNR of intermediate frames is still very higher than H.264 up to 4dB. The average Y-PSNR of SVR with class is around 29.11dB which is over 3dB higher than average PSNR (25.49dB) of H.264. SVR without class which has only T5 input features is performed in order to reduce complexity. If there are not available neighbor MBs, corresponding MBs are considered zeros. The average PSNR of SVR without class is 27.98dB which is over 2dB higher than H.264.

5. CONCLUSION

In this paper, we apply SVR to improve the performance of H.264 intra prediction. Experimental results show that sta-

tistical learning based intra prediction is very promising with high PSNR prediction gain.

6. REFERENCES

- [1] "Advanced video coding for generic audiovisual services," 2003.
- [2] A. C. Yu, G. R. Martin, and H. Park, "A frequency domain approach to intra mode selection in h.264/avc," in *Proc. EUSIPCO*, 2005.
- [3] T. Tsukuba, I. Nagayoshi, T. Hanamura, and H. Tominaga, "H.264 fast intra-prediction mode decision based on frequency characteristic," in *Proc. EUSIPCO*, 2005.
- [4] P. Zhang, D. Zhao, S. Ma, Y. Lu, and W. Gao, "Multiple modes intra-prediction in intra coding," in *Proc. IEEE ICME*, 2004.
- [5] C. Chen, P.-H. Wu, and H. Chen, "Transform-domain intra prediction for h.264," *IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 1479–1500, May 2005.
- [6] T. K. Tan, C. S. Boon, and Y. Suzuki, "Intra prediction by template matching," in *Proc. IEEE ICIP*, 2006.
- [7] L. K. Tang and K. N. Ngan, "Enhancement techniques for intra block matching," in *Proc. IEEE Multimedia and Expo*, 2007.
- [8] J. Yang, B. Yin, Y. Sun, and N. Zhang, "A block-matching based intra frame prediction for h.264/avc," in *Proc. IEEE Multimedia and Expo*, 2006.
- [9] S. Knodo, H. Sasai, and S. Kadono, "Tree structured hybrid intra prediction," in *Proc. IEEE ICIP*, 2004.
- [10] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *NeuroCOLT2 Technical Report NC2-TR-1998-030*, 1998.
- [11] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
- [12] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2002.
- [13] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression: An overview," *IEEE Signal Processing Mag.*, vol. 15, pp. 23–50, Nov. 1998.
- [14] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Mag.*, vol. 15, pp. 74–99, Nov. 1998.
- [15] I. E. G. Richardson, *H.264 and MPEG-4 video compression: video coding for next-generation Multimedia*, John Wiley & Sons Press, 2003.
- [16] T. Evgeniou, M. Pontil, and T. Poggio, *Regularization Networks and Support Vector Machines*, Advances in Computational Mathematics, 2000.
- [17] L. Bo, L. Wang, and L. Jiao, "Recursive finite newton algorithm for support vector regression in the primal," *Neural Computation*, vol. 19, no. 4, pp. 1082–1096, 2007.
- [18] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," 2007.
- [19] S. Lessmann, R. Stahlbock, and S. F. Cron, "Genetic algorithms for support vector machine model selection," in *ICJNN*, 2006.
- [20] Z. Chunhong and J. Licheng, "Automatic parameters selection for svm based on ga," in *Intelligent Control and Automation*, 2004.
- [21] X. Wang, H. Zhang, C. Zhang, X. Cai, J. Wang, and J. Wang, "Prediction of chaotic time series using ls-svm with automatic parameter selection," in *PDCAT*, 2005.
- [22] C. Houck, J. Joines, and M. Kay, "A Genetic Algorithm for Function Optimization: A Matlab Implementation," Tech. Rep. NCSU-IE-TR-95-09, North Carolina State University, Raleigh, NC, 1995.
- [23] R. Liu, E. Liu, J. Yang, M. Li, and F. Wang, *Intelligent Control and Automation: Optimizing the Hyper-parameters for SVM by Combining Evolution Strategies with a Grid Search*, Springer, 2006.
- [24] J. Viron, M. Wien, and H. Schwarz, *Joint Scalable Video Model(JSVM) 11 Software*, Joint Video Team, Doc. JVT-X203, 2007.
- [25] C.-C. Chang and C.-J. Lin, *LIBSVM: A library for Support Vector Machines*, 2007.