Regression-based Intra-prediction for Image and Video Coding

Carlo Noel Ochotorena, Member, IEEE, and Yukihiko Yamashita, Member, IEEE

Abstract—By utilizing previously known areas in an image, intra-prediction techniques can find a good estimate of the current block. This allows the encoder to store only the error between the original block and the generated estimate, thus leading to an improvement in coding efficiency. Standards such as AVC and HEVC describe expert-designed prediction modes operating in certain angular orientations alongside separate DC and planar prediction modes. Being designed predictors, while these techniques have been demonstrated to perform well in image and video coding applications, they do not necessarily fully utilize natural image structures. In this paper, we describe a novel system for developing predictors derived from natural image blocks. The proposed algorithm is seeded with designed predictors (e.g. HEVC-style prediction) and allowed to iteratively refine these predictors through regularized regression. The resulting prediction models show significant improvements in estimation quality over their designed counterparts across all conditions while maintaining reasonable computational complexity. We also demonstrate how the proposed algorithm handles the worst-case scenario of intra-prediction with no error reporting.

Index Terms—intra-prediction, regression, image coding, video coding

I. INTRODUCTION

IMAGES and videos represent vast amounts of digital information. If left uncompressed, this information will correspond to a large storage footprint. Fortunately, a lot of the information in images is redundant. This allows for the use of transforms such as the DCT [1] or DWT [2] to map image blocks into a more compact representation. This process allows a coder to make use of the correlation between neighboring pixels within a block to compress the data. However, this process does not take into consideration neighboring blocks. Intra-prediction schemes make use of these adjacent blocks to create an estimate of the current block to further compact the representation. In particular, previously encoded blocks can be used to form a good estimate of the current block.

A key problem in intra-prediction is finding a good way to estimate pixels from already known pixels. The simplest form of intra-prediction, used in the JPEG standard, is to simply utilize the mean of the previous patch [3]. This form of intra-prediction amounts to no bit cost as only one prediction mode is used exclusively. On the other hand, the prediction, itself, may be quite inaccurate. More recent coding standards such as H.264/AVC [4] and H.265/HEVC [5] make use of more than one prediction mode to estimate the current patch. In such a case, the choice of prediction modes has to be encoded explicitly. There exists a clear trade-off between the number of prediction modes and the quality of the estimate.

Many studies have tackled the task of improving intra-prediction. On one hand, work is being done in efficiently guessing the next prediction mode to be used [6], [7], [8]. If these estimates are correct, then the number of bits needed to describe the next prediction mode can be reduced. Another area of interest is improving the prediction process itself. Many researchers have investigated this particular aspect of intra-prediction leading to a wide array of solutions [9], [10], [11], [12], [13], [14], [15], [16].

In the AVC and HEVC standards, the intra-prediction modes are hand-designed to exploit natural image structures. Often, this is a combination of a zeroth-order DC prediction, a first-order planar prediction, and several angular prediction modes [5]. In contrast, some effort has been made towards learning the predictors, instead [17], [18]. Based on these studies, there is a general improvement that can be obtained through the learning process. We hypothesize that the conventional designed predictors cannot fully utilize image structures as compared to these linear models.

In this work, we present a novel approach to intra-prediction. Our work is similar to [17] and [18] in that it utilizes training patches to train a linear mapping between the known pixels and the pixels to be predicted. However, unlike these approaches, we perform an iterative refinement process similar to that of the k-means clustering approach. We determine which predictor provides the best estimation of each patch and cluster these patches based on the best predictors. A new linear mapping is then calculated for each cluster. As the groupings may shift over time, this process is repeated iteratively. Once this offline learning process is completed, prediction for a single block can be carried out using a single matrix multiplication step.

As we demonstrate in this paper, the proposed algorithm presents several advantages over conventional prediction schemes when applied to images. Under the best conditions, we observe a significant improvement in estimates compared to conventional techniques. More interestingly, we present the results for the worst-case condition where only the top-left block is encoded explicitly and all other blocks are generated through intra-prediction with no error reporting. Through this,
we illustrate a key feature of our approach. The results we report in this paper focus on images for the purpose of isolating the intra-prediction performance. However, our approach is easily extensible towards video coding applications as well.

II. REGRESSION-BASED INTRA-PREDICTION (RIP)

Intra-prediction algorithms depend on previously encoded portions of an image to form an estimate. While this process could theoretically involve all previously decoded pixels and blocks, it is often restricted to the bounding neighbors to reduce complexity (see Fig. 1). In the case of HEVC angular modes, every pixel in the current block is estimated as a weighted average of two pixels [5]. Planar prediction increases this to a four point interpolation problem. DC prediction increases the mode information during the encoding process. However, as we have hypothesized previously, these predictors may not fully exploit image structures. To address this, we group together similar blocks and, instead, attempt to learn the predictors for each group. A trivial manner of grouping these blocks is to make use of the associated predictors \( \hat{p} \) for each patch. It is natural to assume that if a group of patches use the same prediction mode, they are likely to behave similarly. Formally, we define matrices \( X_j \) and \( Y_j \) for the \( j \)-th group:

\[
X_j = \{ x_i \mid \text{argmin}_p \| y_i - M_p x_i \|_2 = j \} \quad (3)
\]

\[
Y_j = \{ y_i \mid \text{argmin}_p \| y_i - M_p x_i \|_2 = j \} \quad (4)
\]

At this point, we utilize the target patches \( y \) and not the estimates \( \hat{y} \). A new predictor can easily be found using least-squares regression:

\[
M_j = \arg\min_M \sum_{x_i \in X_j, y_i \in Y_j} \| y_i - M x_i \|_2 \quad (5)
\]

\[
M_j = Y_j X_j^T (X_j X_j^T + \lambda I)^{-1} \quad (6)
\]

Experimentally, we have found that more stable mappings can be obtained when using an additional Tikhonov regularization term:

\[
M_j = Y_j X_j^T (X_j X_j^T + \lambda I)^{-1} \quad (7)
\]

After all mappings have been learned, the resulting set of predictors should perform better than the original set. However, the patch groupings may also change with this new set of predictors. Intuitively, we can repeat the grouping and training process much like the k-means clustering approach. This allows us to construct progressively improving groups and their corresponding predictors. For all the experiments in this paper, a fixed number of 100 iterations is used. A description of the complete algorithm can be seen in Table 1.

III. EXPERIMENTS

In this work, we carry out two sets of experiments on our proposed Regression-based Intra-prediction (RIP). In the first series of tests, we observe the effect of increasing the number of angular modes. Our second test aims at comparing the RIP mappings with the HEVC prediction modes for varying block sizes. It should be emphasized at this point that our goal is
to characterize the performance of the predictors themselves so coding-specific techniques such as quadtrees are excluded. Likewise, reference sample smoothing and boundary smoothing are not utilized as our primary metric is the error between the original and estimated blocks and not the perceptual quality of the resulting image.

A. Effect of the Number Angular Modes

While the effect of the number of angular modes has been characterized for designed intra-prediction schemes, this may not necessary apply in RIP. To verify this, we manually create a set of angular intra-prediction modes using the two-pixel interpolation scheme in HEVC. For simplicity, we divide the span between 45° and 225° into a set number of uniform angles. Specifically, we test the performance with 5, 9, 13, 17, 21, 25, 29, and 33 angular modes, chosen specifically to preserve the horizontal, vertical, and 45° diagonal orientations.

For the purposes of training, random patches are collected from each image in the Kodak PhotoCD PCD0992 image set. The RIP models are then obtained by applying the proposed refinement algorithm to the original intra-predictors using the training patches to facilitate the regression process. Images outside the training set are used to evaluate the performance of the original and RIP predictors. In particular, we use the well-known Lena, Peppers, and Mandrill images to arrive at a quantitative analysis of the techniques. Without loss in generality, we perform prediction only on the luminance channel of the image.

To evaluate the different predictor sets, we prepared two sets of tests operating on an 8 × 8 block size. A best-case scenario test is performed where all inputs to the patch prediction algorithm are derived from the original data. In this manner, we obtain the best possible prediction assuming lossless encoding of all previous patches. On the opposite end of the spectrum, we test the behavior in the worst possible scenario. If we assume that the encoder has no bits to allocate for the prediction errors, then the image is reconstructed solely using the predictions from the previous patches as well as the information from the top-left patch. While it may be counter-intuitive to include such a test, we would like to demonstrate a unique characteristic of RIP through this.

The results for the best-case tests can be seen in Fig. 2. It is immediately apparent that the RIP estimates are superior across all test samples and in all modes. These results confirm our hypothesis that designed predictors, indeed, are not maximizing the natural image structure. Comparing the performance between the RIP and original estimates, we can observe a mean improvement of 0.757 dB, 0.854 dB, and 0.536 dB for the Lena, Peppers, and Mandrill images respectively.

The worst-case performance results (seen in Fig. 3) demonstrate a surprising behavior. With conventional intra-prediction, pixels are copied and propagated throughout the image. With no error reporting, only pixels from the top-left patch can be propagated. On the other hand, with RIP, the use of regression matrices allows for fluctuations in the prediction values. These fluctuations allow for some form of prediction recovery even in the absence of additional data. This behavior can be more clearly seen in Fig. 4. With a low number of prediction modes, the system accomplishes a smooth but slow-reacting compensation. However, increasing the number of prediction modes allows for the selection of better models even in the most extreme cases of quantization. Compared to the best prediction, we observe an average loss of 5.747 dB,
Comparing prediction performance at evaluate and compare the predictor sets. We also provide data from HEVC are also implemented. The resulting mappings are fed into our algorithm to produce a refined version.

By the standard [5], the DC and planar prediction modes of an HEVC-like prediction scheme. Here, we constructed a prediction matrix with all 33 angular modes as prescribed making it highly useful in low bitrate coding tasks.

Another interesting observation in RIP is the capability to compensate for the lack of information from previous pixels compensating for the increasing dimensionality gap between the reference inputs and the prediction pixels. For instance, in the 32 × 32 case, there are 1024 pixels to be predicted from 97 samples as compared to 64 predictions from 25 samples in the 8 × 8 case.

C. Complexity

An interesting note with regards to RIP is the complexity of the algorithm. Once all regressors have been trained, the intra-prediction task becomes a simple matrix multiplication operation. If we have n pixels to predict from m reference pixels, each group involves mnn multiplications. We can combine the prediction maps by stacking them as:

\[ M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_k \end{bmatrix} \]  

for all k groups. Using this approach, all predictions can be made using a single matrix multiplication involving kmn multiply-accumulate operations. While this approach is more computationally expensive than conventional intra-prediction, it has the advantage of being consistent. No special treatment is needed for any of the patches or any of the prediction modes. In addition, packages such as BLAS [19] exploit optimizations to perform matrix multiplications [20], [21]. Modern hardware are often equipped to efficiently handle matrix-vector multiplications as well [22], [19].

Another interesting observation in RIP is the capability to construct a good estimate of the image in the worst-case scenario. This would allow for a low-complexity coder that performs a serial sweep without error reporting. At this point, additional bits may then be transmitted describing the error between the worst-case reconstruction and the target image. Through such an approach, we completely bypass the RD decisions needed during the intra-coding step. The resulting errors can easily be divided into non-overlapping blocks and processed in parallel, leading to further computational reductions. While this approach is certainly sub-optimal, it nonetheless presents an interesting alternative to the traditional encoding scheme.

IV. Conclusion

In this work, we formulated the intra-prediction task as a mapping problem between input reference pixels and the pixels to be predicted. In such a manner, we constructed an algorithm designed to refine an existing intra-prediction scheme. This refinement is performed iteratively using a regression-based approach leading to a significant improvement in the prediction quality compared to the HEVC prediction scheme. An interesting side effect of the RIP is that it is capable of compensating for the lack of information from previous pixels making it highly useful in low bit rate coding tasks.

REFERENCES

[1] N. Ahmed, T. Natarajan, and K. Rao, “Discrete cosine transform,” vol. C-23, no. 1, pp. 90–93.
[2] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transforms,” vol. 1, no. 2, pp. 205–220.
[3] G. Wallace, “The JPEG still picture compression standard,” vol. 38, no. 1, pp. xviii–xxxiv.
[4] G. Sullivan and T. Wiegand, “Video compression - from concepts to the H.264/AVC standard,” vol. 93, no. 1, pp. 18–31.
[5] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, “Intra coding of the HEVC standard,” vol. 22, no. 12, pp. 1792–1801.
Fig. 7. Sample predictions using 5, 9, 13, 17, 21, 25, 29, and 33 angular modes (left to right) using an $8 \times 8$ block size. Illustrated from the top row to the bottom row respectively: best-case conventional prediction; best-case RIP prediction; worst-case RIP prediction.

[6] J.-C. Wang, J.-F. Wang, J.-F. Yang, and J.-T. Chen, “A fast mode decision algorithm and its VLSI design for H.264/AVC intra-prediction,” vol. 17, no. 10, pp. 1414–1422.

[7] L. Zhao, L. Zhang, S. Ma, and D. Zhao, “Fast mode decision algorithm for intra prediction in HEVC,” in Visual Communications and Image Processing (VCIP), 2011 IEEE, pp. 1–4.

[8] G. Laroche, J. Jung, and B. Pesquet-Popescu, “Intra coding with prediction mode information inference,” vol. 20, no. 12, pp. 1786–1796.

[9] C. An and T. Nguyen, “Statistical learning based intra prediction in H.264,” in Image Processing. 2008. ICIP 2008. 15th IEEE International Conference on, pp. 2800–2803.

[10] J. Balle and M. Wien, “Extended texture prediction for H.264/AVC intra coding,” in Image Processing, 2007. ICIP 2007. IEEE International Conference on, vol. 6, pp. 93–96.

[11] Y. Dai, Q. Zhang, A. Tourapis, and C.-C. Kuo, “Efficient block-based intra prediction for image coding with 2D geometrical manipulations,” in Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, pp. 2916–2919.

[12] F. Kamisli, “Intra prediction based on Markov process modeling of images,” vol. 22, no. 10, pp. 3916–3925.

[13] S. Kondo, H. Sasai, and S. Kadono, “Tree structured hybrid intra prediction,” in Image Processing, 2004. ICIP ’04. 2004 International Conference on, vol. 1, pp. 473–476.

[14] Y. Piao and H. Park, “Adaptive interpolation-based divide-and-predict intra coding for H.264/AVC,” vol. 20, no. 12, pp. 1915–1921.

[15] Y. Sun, M. Xu, X. Tao, and J. Lu, “Online dictionary learning based intra-frame video coding via sparse representation,” in Wireless Personal Multimedia Communications (WPMC), 2012 15th International Symposium on, pp. 16–20.

[16] Y. H. Tan, C. Yeo, Z. Li, and S. Rahardja, “Intra coding with adaptive partial reconstruction,” vol. 23, no. 2, pp. 270–279.

[17] D. Garcia and R. de Queiroz, “Least-squares directional intra prediction in H.264/AVC,” vol. 17, no. 10, pp. 831–834.

[18] L. Zhang, X. Zhao, S. Ma, Q. Wang, and W. Gao, “Novel intra prediction via position-dependent filtering,” vol. 22, no. 8, pp. 687–696, emerging Techniques for High Performance Video Coding.

[19] S. Kestur, J. Davis, and O. Williams, “Bias comparison on FPGA, CPU and GPU,” in VLSI (ISVLSI), 2010 IEEE Computer Society Annual Symposium on, pp. 288–293.

[20] R. Agarwal, F. Gustavsson, and M. Zubair, “A high-performance matrix-multiplication algorithm on a distributed-memory parallel computer, using overlapped communication,” vol. 38, no. 6, pp. 673–681.

[21] M. Potkonjak, M. Srivastava, and A. Chandrakasan, “Multiple constant multiplications: efficient and versatile framework and algorithms for exploring common subexpression elimination,” vol. 15, no. 2, pp. 151–165.

[22] V. Kumar, S. Joshi, S. Patkar, and H. Narayanan, “FPGA based high performance double-precision matrix multiplication,” in VLSI Design, 2009 22nd International Conference on, pp. 341–346.