# TEMPLATE MATCHING FOR IMAGE PREDICTION: A GAME-THEORETICAL APPROACH

*W. Sabrina Lin, Yang Gao, and K. J. Ray Liu*

ECE Dept., University of Maryland, College Park, MD 20742 USA

**ABSTRACT**

This paper presents a game-theoretical approach to provide a framework for optimal template selection in image prediction. Image prediction is an effective tool for coding still images and intra pictures in videos. Template matching algorithms which use neighboring blocks of the prediction target as templates have been widely used for image prediction. The assumption of these approaches is that the template has similar textural structures as the prediction target. Up to now these approaches all use pre-fixed templates for all prediction targets. However, in real images, these fixed templates are very likely to contain textures that are not or are not significant in the prediction targets and these insignificant textures introduce larger prediction residues. In this paper, we propose a coalitional game in which every pixel is treated as a player and tries to seek partners to form a coalition to capture the textural structure. By forming a coalition, every player in the coalition can obtain a gain of improving the ability of capturing the textural structure of coalition while incurring a cost of introducing textural variance within the coalition. Experimental results show that the proposed game-theoretical approach outperforms the conventional pre-fixed template matching prediction up to 2dB coding gain.

## 1. INTRODUCTION

Image prediction technique is an effective method for minimizing the encoded information of an image or an intra frame in a video sequence. Spatial domain intra prediction first appeared in [1] and developments based on spatial domain prediction has been evolved into the H.264/AVC standard [2]. H.264/AVC standard has two prediction types: Intra-16x16 and Intra-4x4. The Intra-16x16 type supports four intra prediction modes while the Intra-4x4 type supports DC mode and eight directional modes. Each 4x4 block is predicted from prior encoded samples from spatially neighboring blocks (directional mode) or the mean of neighboring pixels (DC mode). The directional prediction is done by simply propagating the pixel values along the specified direction. Other enhancement methods based on the same spatial prediction idea such as [3], [4] where more directions and modes were employed. These prediction approaches are suitable in presence of contours, when the directional mode can be chosen corresponding to the orientation of the contour. However, in most cases, it fails in areas with more complex textures.

Template matching is a simple and effective method for texture synthesis [5], and template matching has been used for combating the difficulties of intra prediction with complex textures [6, 7]. In this method, the block to be predicted (can be 4x4 or 8x8 block) is further divided into smaller sub-blocks. The blocks with known value surrounding the prediction target sub-block are considered as "template" for the sub-block. Then the encoder search all over the known areas of the image, i.e., the candidates, to match the template blocks. The matching criteria is to minimize the sum of absolute distance between the template and the candidate. The same procedure is repeated for all four target sub-blocks, and the four best match candidate sub-blocks constitute the prediction of the prediction target block.

There are many improved methods based on template matching. The work in [7] averaged multiple template matching predictors, including larger and directional templates, resulting in more than 15% coding efficiency in H.264/AVC codec. Sparse approximation such as matching pursuit [8] and orthogonal matching pursuit [9] have been proposed to change the matching between the template and candidate into the distance between template and the linear combination of candidates. Also, other matching criterions such as residue coding cost combining with mean square error (MSE) have also been used to replace the sum of absolute distance between template and candidates [10].

The above existing work all based on fixed templates such as the pixels surrounding the prediction target or just pixels at one direction of the target. Even in some approaches the encoder can select from the among several pre-defined templates, those templates are still fixed for all prediction targets. However, how to adaptively choose optimal neighborhoods is also very important since the optimal coding performance will be achieved if the template is consistent with the prediction target in terms of textural structures. If the template contains too less texture structures, the predicted block will be too smooth. On the other hand, if the template contains too many textures that are not significant in the prediction target, the matching will lead to candidates that are too different from the prediction target. And since every prediction target has different textures, the optimal template for each prediction target should be different. Also, if both detector and the encoder adopt the same algorithm of searching optimal template, the template type does not need to be transmitted and hence reduce the coding rate.

In this paper, we propose a general game-theoretical model of segmenting the neighboring area into similar-texture regions and use the regions that are closest to the prediction target as the template. In the game, every pixel is treated as a player, who tries to seek partners to form a coalition to reduce the variance within a coalition while paying the cost of increasing bias. The prediction and coding PSNR/bit-rate performance curves show a gain up to 2 dB when compared with the pre-fixed template matching based prediction.

The rest of this paper is organized as follows. In Section 2, we describe the spatial prediction problem. Then, we show in details the proposed optimal template selection game and the procedure of forming coalition based on local information in Section 3. Finally, we show the experimental results in Section 4 and draw conclusions in Section 5.

The authors can be reached at {wylin, yanggao, kjrliu}@umd.edu.

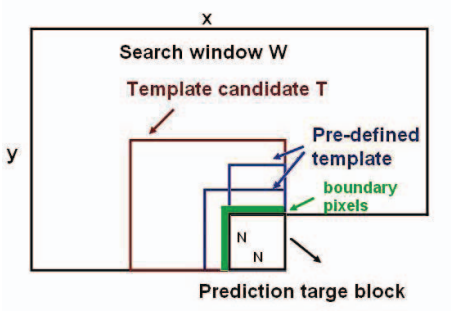



**Fig. 1**: Illustration of the template matching problem

**Fig. 2**: Optimal template for different prediction target blocks of Lena image

## 2. SYSTEM MODEL

Figure 1 illustrates the template-matching based spatial prediction problem. All pixels in the casual search window $W$ are known values, and the only unknown area is the NxN prediction target block. As shown in Figure 1, an example of the pre-defined template is the pixels surrounding the prediction target block, and the other example template is composed of the pixels above the target block. The principle of the template-matching prediction approach is to first search within the search window $W$ for the best approximation or reconstruction for the template, and keep the same procedure to approximate the unknown pixel values in the prediction target. The search window should be casual to ensure the decoder can follow the same scheme to decode the predicted block.

However, the most-representing template of different prediction target block will be different. For example, Figure 2 shows the templates yielding best prediction result for different target blocks, respectively. We can see that from Figure 2, the best templates contain same texture structures as the target block. If the template does not have the most significant texture of the target block, the predicted block will be too smooth. On the other hand, if the template contains too many textures that are not significant in the prediction target, the matching will lead to candidates that are too different from the prediction target.

Therefore, in order to optimally locate template for different prediction targets, we propose to divide the template candidate area $T$ as shown in Figure 1 into segments containing structural similar pixels and choose the most significant segments as the template for prediction.

## 3. GAME-THEORETICAL APPROACH OF TEMPLATE SELECTION

As presented in the previous section, if the template has similar texture as in the prediction target, the template-matching-based prediction schemes can achieve optimal performance. Due to the absence of the prediction target, the best we can do is to predict the texture in the prediction target based on neighboring pixels. Here we propose to divide the neighboring pixels into segments where each segment represents one type of texture. Then we will select the segments with textures that are most likely to be in the prediction block, and use the union of these segments as the template for prediction.

### 3.1. Utility function and solution to the coalitions

The first step of template selection is the divide the template candidate area $T$ as in Figure 1 into partitions, i.e., $T = \{T_1, T_2, ..., T_K\}$. Since the the number of partitions $K$ is unknown, the traditional segmentation and clustering methods may not work. The partition problem can be thought as each pixel trying to find the best partition so that the texture within each partition is consistent. From each pixel's point of view, it has multiple choices of which partition to join, and these partitions are composed of other pixels also. Therefore, each pixel's choice influence the decision of other pixels' decisions and performances, and such complex interactions and dynamics can be modelled as a coalitional game [11–13].

By formulating the segmentation problem as a game, every pixel is treated as a player, and each player tries to seek partners to form coalitions which have consistent texture within each coalition. First, the term texture is an aggregative term, i.e., if a coalition has more pixels with the same pattern, it can represent a texture better. Also, if the coalition already has enough number of pixels, adding in one more pixel will have less improvement of representing a texture. Therefore, the gain of joining a coalition is a concave function of the size of the coalition. Here we use a simple reciprocal function

$$g(T_i) = -\frac{\lambda}{|T_i|}, \tag{1}$$

where $|T_i|$ denotes the size of the partition $T_i$ and $\lambda$ is the balance parameter between cost and reward.

On the other hand, the pixel aims to join the coalition which has the most similar texture as the pixel and its neighborhood. Therefore, the cost of a pixel joining the coalition can be considered as the extra texture variation that the pixel introduces. Such a formulation encourages $T_i$ to be composed of pixels with the same texture. The cost function of forming coalition $T_i$ is

$$c(T_i) = |T_i| * \sigma_{T_i}, \tag{2}$$

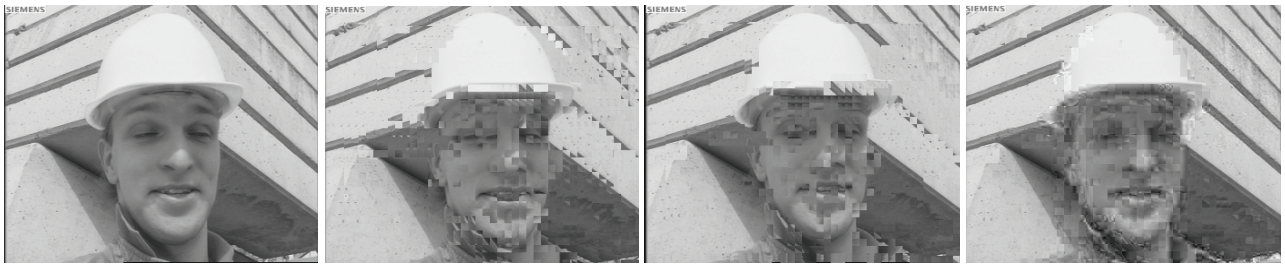
where $\sigma_{T_i}$ is the variance within the coalition $T_i$. Note that the distance metric between two pixels $(x_j, y_j)$ and $(x_k, y_k)$ in $T_i$ is calculated based on the similarity between the patches centering each pixel, respectively. The distance metric can be written as

$$d(j,k) = \sum_{m,n \in W_P} (v(x_j+m, y_j+n) - v(x_k+m, y_k+n))^2, \tag{3}$$

where $W_P$ is the patch window, and $v(x, y)$ is the value of pixel $(x, y)$.

The utility function can then be defined as gain minus cost

$$\pi(T_i) = -\frac{\lambda}{|T_i|} - |T_i| * \sigma_{T_i}. \tag{4}$$



(a) Original image (b) Static template (c) Dynamic template (d) Optimal template

**Fig. 3**: Visual quality comparison of predicted Foreman image with MSE as matching criteria: (a) original image (b) static template (24.01 dB/0.90 bpp), (c)Dynamic template (25.46 dB/ 0.84 bpp), and (d)Optimal template(26.4dB/0.77bpp)

Note that given the above utility function, we can see that when the size of the coalition $|T_i|$ increases, the existing players in the coalition can obtain gains from having more pixels to represent the texture. On the other hand, this gain is limited by the cost which is the increment of total variance within the coalition. Note that the gain is independent of the texture of the joining pixel, therefore, each coalition will always welcome new pixels with most-similar texture. The problem now is to find the optimal coalition structures based on the utility function in (4). This problem can be solved by merge and split rule [11] but it is NP-complete. Since we are only searching within the template candidate set $T$ and not the whole image, the merge and split rule can be applied to solve the unique solution.

### 3.2. Selecting most relevant segments

Now we have already divided the template candidate set $T$ into segments $T_i$, $1 \leq i \leq K$ and each $T_i$ represents one specific texture. The next step is to select a union of segments that contains textures which are most likely to appear in the prediction target block.

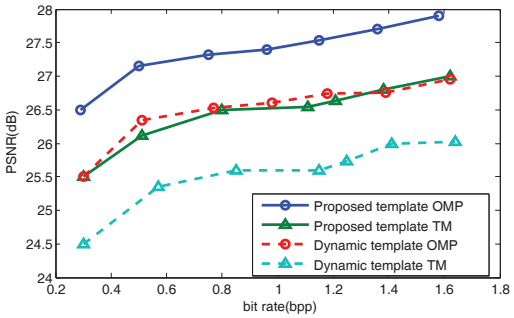
The idea of selecting proper segments is as follows. First, the selected segment $T_i$ should be connected to the prediction target thus it should contain the boundary pixels as indicated in Figure 1. Next, if the prediction target cuts into the texture region of which the remaining parts are in $T_i$, then adding some pixels of the prediction block into $T_i$ will reduce the variance of the locations of pixels in $T_i$. Finally, if the extended boundary of $T_i$ cuts into the prediction target, then the prediction block is likely to contain the texture structure as in $T_i$.

Our template selection procedure can be organized as follows. First set the template to be empty, then:

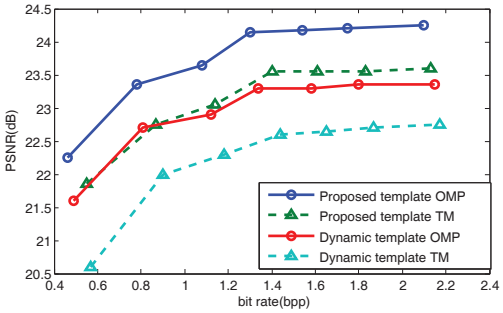
- Let $T'$ be a subset of $T$ and each partition $T_i \in T'$ contains at least one boundary pixel of the prediction target.
- If there exists a pixel $(x_j, y_j)$ in the prediction block such that the location variance of $T_i \in T'$ is larger than that of $T_i \cup (x_j, y_j)$, then $T_i$ is included in the template. Here the location variance $L_{T_i}$ is calculated as

$$L_{T_i} = \sum_{(x_j,y_j)\in T_i} \frac{((x_j, y_j) - (\mu_x, \mu_y))^2}{|T_i|}, \qquad (5)$$

  where $\mu_x = \sum_{(x_j,y_j)\in T_i} x_j/|T_i|$, and $\mu_y = \sum_{(x_j,y_j)\in T_i} y_j/|T_i|$.
- Extend the boundary of $T_i$ according to the gradient near the boundary pixels of the prediction target. If these boundaries cross and form extensions of $T_i$ within the prediction target, then $T_i$ is included in the template.




(a) Foreman




(b) Barbara

**Fig. 4**: PSNR of the predicted signal

## 4. EXPERIMENTAL RESULTS

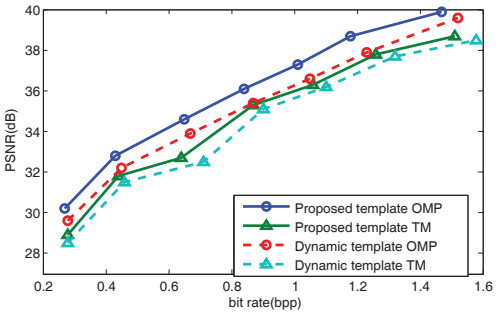
To evaluate the prediction performance of the optimal templates, we first compare the visual qualities of our algorithm with static template matching [6] and dynamic template matching [3] as in Figure 3. The test images are 512x512 Barbara and Foreman in QCIF format. To initialize image prediction, we first intra coded the top 3 rows and left 3 columns of blocks of size 8 8 are with JPEG. Then for each block, the template candidate area is set to be 3 blocks wide an 3 blocks tall surrounding the target block. After determining the optimal template, we match the template with all candidates in the search window which is 9 blocks wide and 5 blocks tall. The match criteria is set to minimize MSE between the template and the matching candidate. When a block has been predicted, the residue is quan-

tized and encoded as JPEG with which a uniform quantization matrix of step size 16 is weighted by a quality factor. The reconstructed image is obtained by adding the quantized residue to the prediction. Since for the decoder can follow the same algorithm to search for the optimal template and then search for the best match, the only extra information need to be transmit is the tradeoff parameter $\lambda$ which is set to be 2.893 in our experiments. Since $\lambda$ is the same for the whole image, the extra amount of information to be transmitted is very small.

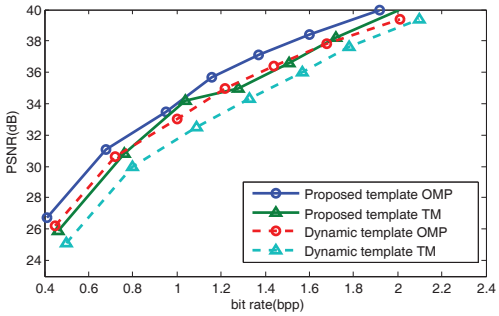
From Figure 3 we can clearly see that the proposed optimal template selection outperforms the conventional pre-defined template scheme, either static template or dynamic template, in visual quality, PSNR and bit rate. Especially, Figure 3(d) preserve the textured area such as the strong texture of the building above the foreman's head as well as the fine small textures on the face.

Next, we will show that our template selection algorithm can be applied to other template-based prediction algorithms. Other than template matching with MSE as the distance metric, we also apply our template selection onto the orthogonal matching pursuit (OMP) [9] algorithm. Since the residue-distortion dynamic template selection usually outperforms static template selection, here we only compare our optimal template selection algorithm with residue-distortion dynamic template selection. In this experiment, we test both Foreman and Barbara with quality factors from 10 to 90.

Figure 4 shows the prediction performance as in PSNR of the predicted images. It is obvious that our proposed algorithm is a basic tool for template selection and when applied to both matching schemes, the quality of the predicted signal is significantly improved in both PSNR and bitrate. Our optimal template selection has at least 1 dB improvement over the conventional pre-fixed template algorithm. The coding performance of both images are demonstrated in Figure 5. We can see that the coding gain of up to 2 dB can be achieved for both images.




(a) Foreman




(b) Barbara

**Fig. 5**: PSNR of the encoded signal

## 5. CONCLUSION

In this paper, we propose a game-theoretical framework to find optimal templates for template-based image intra prediction. The proposed algorithm aims to locate the template with the most significant textures in the prediction target's pre-decoded neighborhood. Each pixel in the neighborhood seeks to form coalitions to minimize the textural difference within a coalition as well as maintaining the size of the segment to better represent the texture. After segmenting the neighboring pixels into single-structure segments, we then determine the template by choosing the segments that are most likely to have the same structure as the prediction target block. The experimental results on real images demonstrate that the proposed template selection can improve the prediction performance over conventional pre-fixed templates in visual quality, PSNR, and bit rate. Also, when combined with existing coding schemes, the our template selection algorithm can provide up to 2 dB coding gain.

## 6. REFERENCES


[1] Gisle Bjontegaard, ""Coding improvement by using 4x4 blocks for motion vectors and transform"," in *ITU-T/Study Group 16 / Video Coding Experts Group*, 1997.

[2] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[3] P. Zhang, D. Zhao, S. Ma, Y. Lu, and W. Gao, "Multiple modes intra-prediction in intra coding," in *Multimedia and Expo, 2004. ICME'04. 2004 IEEE International Conference on*. IEEE, 2005, vol. 1, pp. 419–422.

[4] Z. Nan, Y. Baocai, K. Dehui, and Y. Wenying, "Spatial prediction based intra-coding," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME2004), Taipei*, 2004.

[5] M. Ashikhmin, "Synthesizing natural textures," in *Proceedings of the 2001 symposium on Interactive 3D graphics*. ACM, 2001, pp. 217–226.

[6] T.K. Tan, C.S. Boon, and Y. Suzuki, "Intra prediction by template matching," in *2006 IEEE International Conference on Image Processing*. IEEE, 2007, pp. 1693–1696.

[7] T.K. Tan, C.S. Boon, and Y. Suzuki, "Intra prediction by averaged template matching predictors," in *Consumer Communications and Networking Conference, 2007. CCNC 2007. 4th IEEE*. IEEE, 2007, pp. 405–409.

[8] M. Turkan and C. Guillemot, "Sparse approximation with adaptive dictionary for image prediction," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2010, pp. 25–28.

[9] M. Turkan and C. Guillemot, "IMAGE PREDICTION: TEMPLATE MATCHING vs. SPARSE APPROXIMATION," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 1693–1696.

[10] S. Mallat and F. Falzon, "Analysis of low bit rate image transform coding," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 1027–1042, 1998.

[11] D. Ray, *A game-theoretic perspective on coalition formation*, Oxford University Press, USA, 2007.

[12] Y. Chen and K.J.R. Liu, "A Game Theoretical Approach For Image Denoising," in *IEEE International Conference on Image Processing*, 2010.

[13] M.J. Osborne and A. Rubinste, *A Course in Game Theory*, The MIT Press, 1994.