Gaussian Process Regression Based Prediction for Lossless Image Coding

Wenrui Dai* and Hongkai Xiong*

*Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

E-mail: {daiwenrui, xionghongkai}@sjtu.edu.cn.

Abstract

LS-based adaptation cannot fully exploit high-dimensional correlations in image signals, as linear prediction model in the input space of supports is undesirable to capture higher order statistics. This paper proposes Gaussian process regression for prediction in lossless image coding. Incorporating kernel functions, the prediction support is projected into a high-dimensional feature space to fit the anisotropic and nonlinear image statistics. Instead of directly conditioned on the support, Gaussian process regression is leveraged to make prediction in the feature space. The model parameters are optimized by measuring the similarities based on the training set, which is evaluated by combined kernel function in the sense of translation and rotation invariance among supports mapped in the feature space. Experimental results show that the proposed predictor outperforms most benchmark predictors reported.

1 Introduction

Linear prediction is prevailing in predictive coding of natural images under the assumption of stationary Gaussian random process. Among which, least-square (LS) autoregression based predictors significantly improve the predictive performance, which adapt themselves on a pixel-by-pixel basis to fit the varying local statistics. Traced back to [1], LS-based adaptation optimized coefficients for prediction under the piecewise autoregressive (PAR) assumption for image signals. Moreover, edge-directed prediction (EDP) in [2] figured out the edge-directed property of LS-based adaptation which inspired the LS optimization exactly in the edge areas [3, 4]. For further improvement, two-pass prediction schemes [5, 6] enabled mixture distribution and global image analysis to improve prediction from LS-based adaptation.

Under the assumption of spatial dependencies among neighboring pixels, the prediction support of LS-based adaptation is a fixed-size rectangular causal region centered at the current pixel. However, it ignores varying 2-D spatial statistics of the anisotropic image signals, especially in edge and texture regions. Establishing isotropic Gaussian random field for image signals, [7] analyzed the influence of the pixel scan order on the support to find the optimal order in prediction. However, the isotropic formulation is still restrictive. As an alternative, [8] and [9] utilized weighted least squares (WLS) based on the relative amount of information in the support to adjust the eigenvalues of covariance matrix. [10] optimized pixel ordering for context tree

The work was supported in part by the NSFC, under grants U1201255, 61271218, and 61228101.

1068-0314/14 $31.00 © 2014 IEEE
DOI 10.1109/DCC.2014.72

modeling in arithmetic coding of color map images. To adapt supports to the varying statistics, [11] selected the size of square support according to local image statistics. While [12] suggested an elliptical support and aligned its principal axis with the direction of local edge. Furthermore, [13] adopted the minimum description length (MDL) principle to select supports based on their correlations with the current pixel.

Commonly, LS-based adaptation is a linear model over the support with its parameters optimized based on the training set. However, LS-based linear predictors can only measure the second order moments, but cannot account for the higher order statistics [14]. As a result, the performance of predictive coding is hampered, as the high-dimensional correlations in image signals are not fully exploited, especially in edge and texture regions.

In this paper, we propose Gaussian process regression (GPR) for prediction in lossless image coding. In order to fit the non-stationary image statistics, the prediction support is nonlinearly projected into a high-dimensional feature space using the kernel functions. Consequently, a linear model is applied in this space instead of directly on the support. The model parameters for prediction are trained by relating them to the similarities among the supports for training sets and current pixels. These similarities are measured in terms of translation and rotation invariance in the feature space.

Actually, LS-based adaptation is a restricted kind of Gaussian process regression with linear model and free noise in the input space spanned by its support [15]. Therefore, the similarities among the supports for training sets and current pixels are measured with their inner products in the input space. In the view of similarity, improved methods for selecting support in LS-based adaptation exclude the redundant pixels in support for similarity measure, but do not develop optimal measurement for structures with higher dimension. In contrast, the proposed predictor leverages Gaussian process to make pixel-by-pixel prediction in a nonlinearly projected feature space, which exploits high-dimensional correlations in anisotropic image signals.

To validate the proposed GPR-based adaptive predictor, we apply it to lossless image coding. In the projected feature space, rational quadratic and polynomial dot product function are proposed to measure similarities in the sense of translation and rotation invariance, respectively. The parameters of the two functions are estimated by the automatic relevance determination (ARD) algorithm, which is adaptive to the varying training set neighboring to the current pixels. Remarkably, the encoding and decoding can be performed in one-pass coding without any side information, as the proposed predictor utilizes the causal neighbourhoods as the supports of current pixels. The proposed predictor outperforms both the typical LS-based predictors and the two-pass predictive coding schemes, e.g. MRP and TMW.

The rest of the paper is organized as follows. Section 2 describes the Gaussian process regression based prediction for lossless image coding. In Section 3, kernel functions are developed for similarity measure in the sense of translation and rotation invariance in the projected feature space. Experimental results for validation are reported in Section 4. Section 5 concludes this paper.

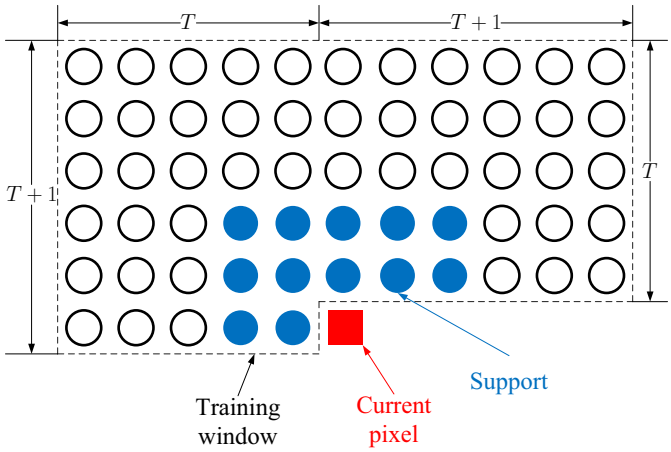


Figure 1: GPR-based predictor of order 12 and the corresponding training window used to optimize the prediction coefficients.

2 Gaussian Process Regression Based Prediction

In this section, we propose the GPR-based prediction, including its formulation and the estimation of its parameters. In the view of Gaussian process regression, LS-based adaptation is found to be one of its special cases with the dot product kernel function.

2.1 Formulation of Gaussian Process Regression

Gaussian process (GP) is a powerful, non-parametric tool for learning regression functions from the sample data. Key advantages of Gaussian process are modeling flexibility, capability to provide uncertainty estimates, and ability to learn noise and smoothness parameters from the training set. Given the current pixel y_* for predicting and its support $\mathbf{x}_*$, we assume that the training set $\mathcal{S} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ is composed of N couples of an M-tuple support $\mathbf{x}_i = [x_{i1}, x_{i2}, \cdots, x_{iM}]^T$ and a scalar output value y_i drawn from

$$y_i = f(\mathbf{x}_i) + \epsilon, \tag{1}$$

where the term ϵ is white noise with variance σ_n^2. As shown in Fig. 1, y_i is the predicted pixels in the training window with a size of $2T(T+1)$ and $\mathbf{x}_i$ is its support that has similar causal neighborhoods as the current pixel. For convenience, both the vectorial supports and the predicted pixels are collected in the form of $X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ and $\mathbf{y} = [y_1, y_2, \cdots, y_N]^T$, respectively. Under the assumption of Gaussian process, the collected vector $\mathbf{y}$ of the predicted pixels follows a zero-mean multivariate Gaussian distribution.

$$\mathbf{y} \sim \mathcal{N}\left(0, K(X, X) + \sigma_n^2 I\right), \tag{2}$$

where I is the identity matrix, and $K(X, X)$ is the covariance matrix with its elements determined by $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. The kernel function $k(\mathbf{x}, \mathbf{x}')$ is designed to measure the similarity between two supports $\mathbf{x}$ and $\mathbf{x}'$, which is commonly associated with a diagonal scale matrix $\mathbf{W}$. Here, the term $\sigma_n^2 I$ introduces the Gaussian noise derived from ϵ in Eq. (1).

When the observation is noise-free, the prediction of the multivariate Gaussian distribution is simplified by obtaining the covariance matrix conditioned on the training set. The current pixel y_* is predicted conditioned on the support $\mathbf{x}_*$ and the training set $\mathcal{S}$ is obtained by estimating the mean and variance of the multivariate Gaussian distribution. In detail, the mean μ is estimated by

$$\mu\left(\mathbf{x}_*, X, \mathbf{y}\right) = K\left(\mathbf{x}_*, X\right) K\left(X, X\right)^{-1} \mathbf{y}, \tag{3}$$

and the variance Σ is

$$\Sigma\left(\mathbf{x}_*, X, \mathbf{y}\right) = K\left(\mathbf{x}_*, \mathbf{x}_*\right) - K\left(\mathbf{x}_*, X\right) K\left(X, X\right)^{-1} K\left(X, \mathbf{x}_*\right), \tag{4}$$

where $K\left(\mathbf{x}_*, X\right)$ measures the similarity between $\mathbf{x}_*$ and X and is Abelian. Consequently, the mean μ is taken as the prediction of the current pixel y_*.

In fact, the kernel functions are commonly characterized by a set of parameters $\theta = [\mathbf{W}, f]$. They are learned by pursuing the maximized conditional log-likelihood based on the training set.

$$\hat{\theta} = \arg\max_{\theta} \left\{\log p\left(\mathbf{y}|X, \theta\right)\right\} \tag{5}$$

Eq. (5) can be iteratively implemented by using numerical optimization techniques, as described in Section 3.3.

2.2 Kernel Function for LS-based Adaptation

This section analyzes the LS-based adaptation in the view of Gaussian process regression, where each pixel is predicted based on its support obtained from its causal neighborhood in a raster scan order in the sense of Euclidean distance. Taking consideration of the 2-D spatial dependencies in image signals, the training set $\mathcal{S}$ is similarly collected as shown in Fig. 1. Consequently, the model parameters are adaptively estimated by locally seeking a minimized squared error over the training set.

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{y} - X^T\mathbf{w}\|^2 \tag{6}$$

Expanding the squared error and pursuing zero value of the partial derivative of $\mathbf{w}$, the least square estimation $\hat{\mathbf{w}}$ is obtained by

$$\hat{\mathbf{w}} = \left(XX^T\right)^{-1} X\mathbf{y}. \tag{7}$$

Noting the fact that $\left(XX^T\right)X = X\left(X^TX\right)$, the prediction of the current pixel y_* conditioned on its support $\mathbf{x}_*$ and the training set $\mathcal{S}$ is rewritten in the linear form.

$$y_* = \mathbf{x}_*^T\hat{\mathbf{w}} = \left(\mathbf{x}_*^TX\right)\left(X^TX\right)^{-1}\mathbf{y} \tag{8}$$

Obviously, Eq. (8) coincides with the Gaussian process regression with linear function $f\left(\mathbf{x}\right) = \mathbf{x}^T\mathbf{w}$. However, the main drawback is that such linear prediction allows only a limited flexibility. Owing to the non-stationary local statistics of natural images,

the relationship between input X and output $\mathbf{y}$ cannot be reasonably approximated with linear assumption. In the view of Gaussian process regression, the covariance matrix for LS-based adaptation is $K(X, X) = X^T X$, where each element is defined by the kernel function in Eq. (9).

$$k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j \tag{9}$$

Therefore, the kernel function of LS-based linear prediction is isotropic and stationary, which is the dot product of arbitrary two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ with identity scale matrix $\mathbf{W} = \mathbf{I}$. Furthermore, the improvement of support selection is equivalent to utilizing a compact training set $\bar{\mathcal{S}} \subseteq \mathcal{S}$ or adopting anisotropic scale matrix $\mathbf{W}$. The fact shows that the dot product function for covariance matrix is still maintained without exploiting high-dimensional correlations.

3 Kernels for Similarity Measure

Instead of dot product function with identity scale matrix in the LS-based adaptation, we proposed to measure the similarities among supports in the high-dimensional feature space. To maintain the geometry of the input space, similarities are measured in the sense of invariance under certain global transforms, like translation and rotation. Consequently, the eigenvalues for these transforms are constrained, so that the covariance matrix before and after transform can be isomorphic.

3.1 Similarity Measure for Translation Invariance

Appending a constant at the end of support $\mathbf{x}$ and rearranging them, we can obtain the translation transform in a matrix form.

$$T\mathbf{x} = \begin{pmatrix} \mathbf{I} & \mathbf{b} \\ 0 & 1 \end{pmatrix} \mathbf{x}, \tag{10}$$

where $\mathbf{b}$ is a constant vector that determines the displacement for translation. Consequently, the kernel function $k(\mathbf{x}, \mathbf{x}')$ is associated with $\mathbf{x} - \mathbf{x}'$ to keep the covariance matrix translation invariant. From (10), we can obtain

$$T\mathbf{x} - T\mathbf{x}' = \mathbf{I}(\mathbf{x} - \mathbf{x}') = \mathbf{x} - \mathbf{x}'.$$

Such that $k(T\mathbf{x}, T\mathbf{x}') = k(\mathbf{x}, \mathbf{x}')$.

The rational quadratic (RQ) function is a typical kernel function invariant to the translation transform, which is in the form of

$$k(\mathbf{x}, \mathbf{x}') = \left(1 + \frac{(\mathbf{x} - \mathbf{x}')^T \mathbf{W} (\mathbf{x} - \mathbf{x}')}{2\alpha\ell^2}\right)^{-\alpha}. \tag{11}$$

The RQ function is stationary and invariant to all rigid motions. Therefore, it allows for inference from the covariance matrix based on all pairs of sampling supports $\mathbf{x}_i$ and $\mathbf{x}_j$ with the same lag vector $\|\mathbf{x}_i - \mathbf{x}_j\|$. The RQ function can be viewed as

a scale mixture of the squared exponential (SE) covariance functions with different characteristic length scales ℓ.

$$k_{RQ}(r) = \int e^{\frac{r^2}{2\ell^2}} p(\ell)\, d\ell, \tag{12}$$

where $p(\ell|\alpha, \beta)$ is a Gamma distribution with $\beta = \ell^{-2}$ and the shape parameter $\alpha > 0$. Here, the length scale is an important property for the kernel function, which vanishes when the input is greater than the length scale. Thus, a short length scale makes SE covariance function vary rapidly to fit the training set, while the longer one is suitable for slowly-varying signals. Since SE covariance function $exp(-r^2/2\ell^2)$ is with a characteristic length scale ℓ, the RQ function is supposed to weight over all the scales with respect to the Gamma distribution $p(\ell|\alpha, \beta)$. Consequently, it adapts to varying distances $\|\mathbf{x}_i - \mathbf{x}_j\|$ between arbitrary two supports $\mathbf{x}_i$ and $\mathbf{x}_j$.

3.2 Similarity Measure for Rotation Invariance

The rotation transform with an angle θ can be represented in the matrix form.

$$T\mathbf{x} = \begin{pmatrix} \mathbf{R} & 0 \\ 0 & 1 \end{pmatrix} \mathbf{x}, \tag{13}$$

where the rotation matrix is the orthogonal matrix, whose eigenvalues are associated with $\cos\theta \pm i \sin\theta$. Consequently, the dot product of two M-tuple supports holds

$$\langle \mathbf{Rx}, \mathbf{Rx}' \rangle = \mathbf{x}\mathbf{R}^T\mathbf{R}\mathbf{x}' = \mathbf{x}\left(\mathbf{R}^T\mathbf{R}\right)\mathbf{x}' = \langle \mathbf{x}, \mathbf{x}' \rangle.$$

Therefore, the dot product function keeps the rotation transform invariant.

We take the polynomial dot product function $k(\mathbf{x}, \mathbf{x}')$ with degree p as example.

$$k(\mathbf{x}, \mathbf{x}') = \left(1 + \sum_{k=1}^{M} x_k x'_k\right)^p \triangleq \phi_p(\mathbf{x}) \cdot \phi_p(\mathbf{x}') \tag{14}$$

By expanding the polynomial kernel $k(\mathbf{x}, \mathbf{x}')$, its M^p components can be merged and represented by $\binom{M+p}{p}$ kernel bases. Thus, the feature vector $\phi_p(\mathbf{x})$ for $k(\mathbf{x}, \mathbf{x}')$ with degree p is in the form of

$$\left(1, \sqrt{p}x_1, \cdots, \sqrt{p}x_M, \sqrt{\binom{p}{2}}x_1^2, \cdots, \sqrt{\binom{p}{2}}x_M^2, \right.$$
$$\left. \sqrt{2!\binom{p}{2}}x_1x_2, \cdots, \sqrt{2!\binom{p}{2}}x_{M-1}x_M, \cdots, x_M^p \right)$$

$\phi_p(\mathbf{x})$ spans the space of polynomials in $\mathbb{R}^M$ with degree p. According to Weierstrass polynomial approximation theorem, there exists $\phi_p(\mathbf{x}_*)$ to approximate the current pixel y_*, as $p \to \infty$. Since $\phi(\mathbf{x})$ is constructed by the polynomial bases with degree not greater than p, the LS-based adaptation is achieved over the first order bases

$(x_1, \cdots, x_M)$. By tuning p, $k(\mathbf{x}, \mathbf{x}')$ can asymptotically approximate y_* based on the training data in a flexible and refined way. In the regression of the high-dimensional feature vectors, the support $\mathbf{x}$ is projected into $\mathbf{x}'$ by associating their lengths and principal angles. Remarkably, it is non-stationary and invariant when the coordinates are rotated.

3.3 Combined Kernel for Similarity Measure

Since the sum of two kernels is still a kernel [15], various kernel functions can be combined for similarity measure. In the sense of both translation and rotation invariance, the proposed combined kernel function over arbitrary supports $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$k(\mathbf{x}_i, \mathbf{x}_j) = \gamma_1 \left(1 + \frac{(\mathbf{x} - \mathbf{x}')^T \mathbf{W}_1 (\mathbf{x} - \mathbf{x}')}{2\alpha\ell^2}\right)^{-\alpha} + \gamma_2 \left(1 + \mathbf{x}_i^T \mathbf{W}_2 \mathbf{x}_j\right)^p + \gamma_3. \quad (15)$$

Consequently, the parameters for Gaussian process regression are $\theta = \{\gamma_1, \gamma_2, \gamma_3, \alpha, \ell, \mathbf{W}_1, \mathbf{W}_2, p\}$. Thus, they can be iteratively tuned and optimized over the training set based on their gradients $dk(\mathbf{x}_i, \mathbf{x}_j)/d\theta$. Due to space limitation, we do not provide the detailed formulation of these gradients.

Algorithm 1 Gaussian prediction regression based prediction with automatic relevance determination (ARD)

Input: training set $\mathcal{S} = \{\mathbf{x}_i, y_i\}_{i=1}^N$, kernel function k, current support $\mathbf{x}_*$
Output: prediction of current pixel y_*
Initialize parameter θ
Cholesky decomposition L of covariance matrix $K(\mathbf{X}, \mathbf{X})$
if L is invertible **then**
  repeat
    Solve α from equation $LL^T\alpha = \mathbf{y}$
    Calculating log-likelihood $\log p(\mathbf{y}|\mathbf{X})$
    Updating θ with conjugate gradient descent
    Calculating $K(\mathbf{X}, \mathbf{X})$ with updated θ
  until Iterate n times
  Solve α from equation $LL^T\alpha = \mathbf{y}$
  $y_* = k(\mathbf{x}_*, \mathbf{x})\alpha$
else
  y_* is the mean of predicted pixels $\mathbf{y}$
end if
Return y_*

Algorithm 1 shows the GPR-based prediction with automatic relevance determination (ARD). For each pixel to be predicted, Cholesky decomposition is incorporated for its covariance matrix $K(\mathbf{X}, \mathbf{X})$ based on the training set $\mathcal{S}$. Subsequently, the maximum log-likelihood estimation for conditional probability $p(\mathbf{y}|\mathbf{X})$ over $\mathcal{S}$ is solved and the parameter θ is updated with conjugate gradient descent. The solving and updating process is repeated until obtaining the optimal parameters for prediction.

Figure 2: Test image set. From left-top to right-bottom: "Airplane", "Baboon", "Lena", "Peppers", "Balloon", "Barb", "Barb2", "Goldhill", "Couple", and "Cameraman".

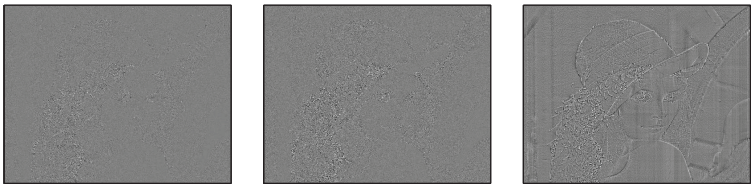
Table 1: Performance comparison with existing lossless image coders in bits per pixel (bpp).

Image(size)	Proposed	MRP	BMF	TMW	Glicbawls	CALIC	JPEG-LS	JPEG 2000
Airplane(512×512)	**3.451**	3.591	3.602	3.601	3.668	3.743	3.817	4.013
Baboon(512×512)	**5.641**	5.663	5.714	5.738	5.666	5.875	6.037	6.107
Balloon(720×576)	**2.544**	2.579	2.649	2.649	2.640	2.825	2.904	3.031
Barb(720×576)	3.821	**3.815**	3.959	4.084	3.916	4.413	4.691	4.600
Barb2(720×576)	**4.184**	4.216	4.276	4.378	4.318	4.530	4.686	4.789
Camera(256×256)	3.964	**3.949**	4.060	4.098	4.208	4.190	4.314	4.535
Couple(256×256)	**3.339**	3.388	3.448	3.446	3.543	3.609	3.699	3.915
Goldhill(720×576)	**4.178**	4.207	4.238	4.266	4.276	4.394	4.477	4.603
Lena(512×512)	**3.880**	3.889	3.929	3.908	3.901	4.102	4.238	4.303
Peppers(512×512)	**4.170**	4.199	4.241	4.251	4.246	4.421	4.513	4.629
Average	**3.917**	3.950	4.012	4.042	4.038	4.210	4.338	4.453

4 Experimental Results

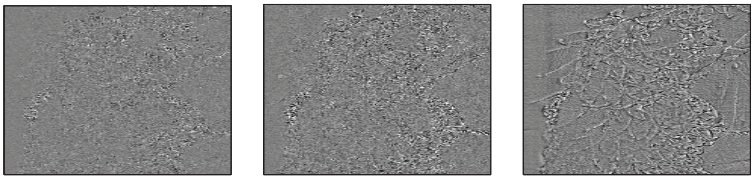
We report the lossless code lengths of the natural images achieved by the proposed GPR-based predictor. For lossless coding, predictions form the proposed predictor are subtracted from the actual pixel values, and the prediction errors are transmitted to the range coder. Both the encoder and decoder operate on a PC with a 3.2GHz Intel Core i7 processor and are complied with VC++ 9.0 with same configuration ("DEBUG" mode). In the experiments, the order M of predictor is 12. In training, the window size T is set to 5 and there are 60 samples in the training set $\mathcal{S}$ for predicting each pixel. For generality of the validation, the selected test images shown in Figure 2 span a wide rage in bit rates.

For validation, we compare the proposed predictor with the existing benchmarks in lossless image coding, as shown in Table 1. Among which, BMF [16] is a special compressor designed for generic images and CG graphs with shapes. CALIC [17] adopted the gradient adjusted predictor (GAP) for context-based adaptive lossless image coding. The proposed predictor outperforms MRP, the best predictor reported, in 8 of 10 test images. In comparison to MRP, it improves the coding performance by an average margin of 0.033 bpp and the coding gain can be up to 3%. In a note of practical interest, there is approximately 10% and 14% enhancement in bit rates for the proposed predictor, when compared with JPEG-LS standard and the JPEG 2000 lossless mode on average. The experimental results show that the proposed



(a) Proposed (b) MRP (c) EDP

Figure 3: Prediction error maps for test image "Lena" obtained by the proposed predictor, MRP, and EDP. Their first order entropies from left to right are 3.944, 4.025, and 4.237 bits per pixel (bpp), respectively.



(a) Proposed (b) MRP (c) EDP

Figure 4: The zoomed detail of "hair" region in "Lena" obtained by the proposed predictor, MRP, and EDP, respectively.

GPR-based predictor improves the predictive coding performance of natural images.

Figure 3 shows the prediction error maps for "Lena", which are obtained by the proposed predictor, MRP, and EDP, respectively. Figure 3(c) shows that the prediction error map by EDP is most evident. While the proposed predictor is more illegible in comparison to MRP, especially in the "hair" region. The zoomed details are shown in Figure 4. It implies that the proposed predictor can improve the predictive results around the texture regions, which are characterized by higher order statistics.

5 Conclusion

Recognizing that LS-based adaptation is a restricted kind of Gaussian process regression with linear model and free noise in the input space of support, we propose Gaussian process regression based predictor for lossless image coding. GPR projects the prediction support into a high-dimensional feature space with the kernel functions in a nonlinear way. Linear prediction is made in the projected feature space, rather than directly conditioned on the support. Its parameters are optimized by considering the similarities among supports for the training set and the current pixel, which is measured with a combined kernel function regarding translation and rotation invariance in the feature space. The proposed GPR-based predictor can fit non-stationary statistics in image signals, as it outperforms the best predictor MRP in most cases.

References

[1] X. Wu, K. Barthel, and W. Zhang, "Piecewise 2D autoregression for predictive image coding," in *Proceedings of the IEEE International Conference on Image Processing*, Chicago, IL, USA, Oct. 1998, pp. 901-904.

[2] X. Li and M. Orchard, "Edge-directed prediction for lossless compression of natural images," *IEEE Transactions on Image Processing*, vol. 10, no. 6, pp. 813-817, Jun. 2001.

[3] L.-J. Kau and Y.-P. Lin, "Adaptive lossless image coding using least-squares optimization with edge-look-ahead," *IEEE Transactions on Circuits and Systems II*, vol. 52, no. 11, pp. 751-755, Nov. 2005.

[4] L.-J. Kau and Y.-P. Lin, "Least-squares-based switching structure for lossless image coding," *IEEE Transactions on Circuits Systems I*, vol. 54, no. 7, pp. 1529-1541, Jul. 2007.

[5] I. Matsuda, N. Ozaki, Y. Umezu, and S. Itoh, "Lossless coding using variable block-size adaptive prediction optimized for each image," in *Proceedings of the European Signal Processing Conference*, Antalya, Turkey, Sep. 2005.

[6] B. Meyer and P. E. Tischer, "TMW-a new method for lossless image compression," in *Proceedings of the IEEE Picture Coding Symposium*, Berlin, Germany, Oct. 1997, pp. 533-538.

[7] N. Memon, D. L. Neuhoff, and S. Shende, "An analysis of some common scanning techniques for lossless image coding," *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1837-1848, Nov. 2000.

[8] B. Meyer and P. E. Tischer, "Glicbawls - Grey level image compression by adaptive weighted least squares," in *Proceedings of the Data Compression Conference*, Snowbird, UT, USA, Mar. 2001, p. 503.

[9] H. Ye, G. Deng, and J. C. Devlin, "A weighted least-squares method for adaptive prediction in lossless image coding," in *Proceedings of the IEEE Picture Coding Symposium*, Saint-Malo, France, Sep. 2003, pp. 489-493.

[10] A. Akimov, A. Kolesnikov, and P. Franti, "Lossless compression of color map images by context tree modeling," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 114C120, Jan. 2007.

[11] C. Kervrann and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image regularization and representation," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 45-69, Aug. 2008.

[12] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstructions," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349-366, Feb. 2008.

[13] X. Wu, G. Zhai, X. Yang, and W. Zhang, "Adaptive sequential prediction of multidimensional signals with applications to lossless image coding," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 36-42, Jan. 2011.

[14] S. Zhu, "Statistical modeling and conceptualization of visual patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 691-712, Jun. 2003.

[15] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, the MIT Press, 2006.

[16] D. Shkarin, A Special Lossless Compressor BMF, Version 2.0 2009. [Online]. Available: http://compression.ru/ds/

[17] X. Wu and N. Memon, "Context-based adaptive lossless image coding," *IEEE Transactions on Communications*, vol. 45, no. 4, pp. 437-444, Apr. 1997.